

NASA Technical Memorandum 85743

**Static and Dynamic Structural-
Sensitivity Derivative
Calculations in the Finite-
Element-Based Engineering
Analysis Language (EAL) System**

FOR REFERENCE

NOT TO BE TAKEN FROM THIS ROOM

Charles J. Camarda and Howard M. Adelman

MARCH 1984

NASA

NASA Technical Memorandum 85743

Static and Dynamic Structural-
Sensitivity Derivative
Calculations in the Finite-
Element-Based Engineering
Analysis Language (EAL) System

Charles J. Camarda and Howard M. Adelman
Langley Research Center
Hampton, Virginia



National Aeronautics
and Space Administration

**Scientific and Technical
Information Office**

1984

INTRODUCTION

Researchers at the Langley Research Center are currently developing and applying the latest optimization capability to the multidisciplinary optimization of aircraft and spacecraft (ref. 1). A key part of this effort is to efficiently calculate structural-sensitivity derivatives which quantify the change in a behavior variable with respect to a structural parameter and which are used in the following applications: (1) to act as input to optimization algorithms; (2) to enhance response analysis programs which aid engineering judgment leading to design modifications; (3) to guide the modification of a finite-element model to better correlate analytical and test results; and (4) to approximate structural response by using Taylor series expansions.

The most basic and straightforward approach to sensitivity analysis is the finite-difference method; however, it is computationally slow and its accuracy must be verified by convergence checks. Analytical methods for calculating exact derivatives from the governing equations have been developed (refs. 2 to 10). These methods greatly reduce the computational effort but are somewhat cumbersome to implement for bending-type elements (ref. 4). Most recently, a semianalytical method for calculating derivatives has been developed which has the generality and programming ease of the finite-difference method while retaining much of the efficiency of the analytical method (ref. 11).

The purpose of this paper is to describe the implementation and verification of the aforementioned methods for computing sensitivity derivatives in the structural finite-element computer program currently used in multidisciplinary optimization studies. The program, denoted the Engineering Analysis Language (EAL) System (ref. 12), is similar to its predecessor, SPAR (ref. 13). It is a modular system of individual analysis processors which may be used in any appropriate sequence to perform a variety of analyses. The EAL System differs from SPAR by providing FORTRAN-like commands which permit branching, testing data, looping, and calling runstreams (similar to calling FORTRAN subroutines). These capabilities permit the implementation of sensitivity calculations without changing the basic program or requiring user-written subroutines in separate programs (as in ref. 4). Further, use of the EAL System avoids the need for extensive operating system control commands as used previously (ref. 4) and thus assures machine independence of the resulting system.

This paper draws on results from references 4, 8, and 11 for the basic methodology and presents EAL input runstreams which calculate derivatives of displacements, stresses, vibration frequencies and mode shapes, and buckling loads and mode shapes with respect to structural variables in the model. The variables are sectional properties including thicknesses, cross-sectional areas, and moments of inertia. Results are presented and comparisons are made among analytical, semianalytical, and finite-difference methods for the following four structural configurations: a swept wing, a box beam, a stiffened cylinder with a cutout, and a space radiometer-antenna truss.

SYMBOLS AND ABBREVIATIONS

A	cross-sectional area
E	modulus of elasticity
$\{F\}$	pseudo load vector
$[F]$	matrix of pseudo load vectors (eq. (21))
f	applied load
$\{f\}$	applied load vector
$[G]$	matrix which relates stresses and temperatures (eq. (26))
$[I]$	identity matrix
I_1, I_2	moments of inertia
J_O	polar moment of inertia
$[K]$	stiffness matrix
$[K_g]$	geometric stiffness matrix
$[M]$	mass matrix
P_j	structural definition parameter
$\{Q\}$	particular solution of equation (35)
r	independent design variable for tube element (eq. (40))
r_i, r_o	inner and outer radii of tube element
$[S]$	stress-displacement matrix
T	temperature
T_O	stress-free temperature
t_i	element thickness
u, v, w	displacements in x , y , and z directions, respectively
$\{u\}$	displacement vector
$\{V\}$	vector of independent design variables
$\{v\}$	vector of dependent design variables
x, y, z	Cartesian coordinates
$\{\alpha\}$	vector of linear thermal expansion coefficients

$\{\lambda\}, \{\omega^2\}$ vectors of buckling loads and vibration frequencies, respectively

ν Poisson's ratio

$[\Phi], [\phi]$ matrices of buckling and vibration mode shapes, respectively

ρ density

$\{\sigma\}$ stress vector

Superscripts:

n new (perturbed)

o original (unperturbed)

Abbreviations:

NDDV number of dependent design variables

NIDV number of independent design variables (NDV in computer listing)

NPOL degree of polynomial used in linking (eq. (1))

NSDP number of structural definition parameters

DESCRIPTION OF THE ENGINEERING ANALYSIS LANGUAGE (EAL) SYSTEM

The EAL finite-element analysis system (ref. 12) evolved from the SPAR computer program (ref. 13). As indicated in figure 1(a), the EAL System contains individual processors which communicate through a data base consisting of one or more libraries of data sets. The data sets typically contain data describing the finite-element model of the structure (e.g., node point coordinates and material properties) as well as response information such as displacements and stresses. All data base communications between processors are in terms of data sets. A set of data-handling utilities transfers data between the processors in central memory of the computer and the data base on auxiliary storage (ref. 14). Contents of individual data sets for SPAR are compiled and listed in reference 15. (In some instances these data sets may be slightly different from corresponding EAL data sets.)

A list of the EAL processors is shown in the uppermost block of figure 1(a). The functions of the various processors are described in table 1. The processors may be executed in any appropriate sequence. A sequence of processor executions is denoted a runstream and may be defined and stored in the data base as a runstream data set. Runstreams and runstream data sets may be nested within an input file.

The EAL system differs from SPAR in its use of a set of flexible FORTRAN-like statements, denoted executive control system (ECS) commands, which allow branching, testing data, looping, and calling runstreams (similar to calling FORTRAN sub-routines). The ECS commands are always preceded by an asterisk (*) and are used to execute processors; for example, the ECS command to execute the TAB processor is *XQT TAB. A list of the ECS commands and their functions is given in table 8-1 of reference 12. The ECS commands are also used to test variables (denoted registers). Registers may be input data, output data, or simple variables defined and manipulated

by the EAL user. If the value of a register is regarded as a variable which changes during an EAL analysis, the current value of the register is defined as a surrogate. A surrogate is always enclosed in quotation marks. Registers and surrogates are defined and manipulated by the use of register action commands (RAC). A RAC is always preceded by an exclamation point (!) in the EAL input. For example, !A=ABS("STR") assigns the absolute value of the surrogate "STR" to register A. A complete list of the register action commands is given in table 7.3-1 of reference 12.

The makeup of a sample EAL input file is illustrated in figure 1(b). The file begins with an ECS command to execute processor TAB using appropriate input data following the command; second, there is a call to execute runstream data set CHNG DV; third, there is execution of processor ELD using appropriate input data; and finally, there is execution of processors of K and INV, neither of which require user input.

The makeup of the sample runstream data set CHNG DV is shown in figure 1(c). The function of this runstream is to loop over the membrane elements in the model (looping is controlled by the JLZ command), extract the membrane thicknesses from data set DV DFN and store them in register DV (using the DS register action command), and finally update that part of the input file in which the membrane thicknesses are defined. The updated portion of the file follows the SA identifier and contains the membrane element numbers ("NI") and the corresponding thicknesses ("DV"). Following the satisfaction of the loop and the arrival at LABEL 2, control returns to the calling runstream by means of the RETURN command.

SCOPE OF CAPABILITY FOR CALCULATING STRUCTURAL-SENSITIVITY DERIVATIVES

Methods Used and Overall Capability

In the present work, structural behavior quantities include displacements $\{u\}$, stresses $\{\sigma\}$, buckling loads and mode shapes $\{\lambda\}$ and $[\Phi]$, and vibration frequencies and mode shapes $\{\omega^2\}$ and $[\phi]$. Structural variables include element thicknesses, cross-sectional areas, and moments of inertia. The following methods for calculating derivatives of structural behavior quantities have been implemented: (1) an analytical method, (2) a semianalytical method (also called the indirect method in refs. 11 and 16), and (3) a finite-difference method (also called the direct method in ref. 11). A list of applicable EAL finite elements is given in table 2. The analytical method was not implemented for the elements which include bending deformation (E32, E33, E42, and E43) because of the algebraic complexity (discussed in the following section).

Design-Variable Definition

Structural modifications are specified as changes to certain structural quantities called design variables, which are related to section properties or mass properties of the finite-element model. A structural definition parameter (ref. 4) is defined to be a parameter which has a linear contribution to the stiffness matrix or the mass matrix, or both, of individual finite elements in the structural model. The design variables can be identical to or have a one-to-one relationship to the structural definition parameters. For example, the cross-sectional area of a rod element is a structural definition parameter, and the areas of several elements in the structure could be equal to a single design variable. In some instances, there is a non-linear relationship between the structural definition parameters and the design

variables, for example, when the moment of inertia per unit width of a plate $I_1 = t^3/12$ is a structural definition parameter and the plate thickness t is a design variable. Finally, in some optimization techniques (e.g., ref. 3), design-variable linking is used to reduce the number of independent design variables. In this instance independent design variables can be linearly or nonlinearly related to the dependent design variables, which in turn can be linearly or nonlinearly related to the structural definition parameters. Chain-rule partial differentiation is then used to compute the required derivatives (ref. 4).

Design-Variable Linking

When linking is used, the number of dependent design variables (NDDV) is larger than the number of independent design variables (NIDV), and the two are related via a mathematical relationship which is often linear (ref. 17). In the present work the following nonlinear design-variable linking algorithm is used:

$$v_i = C_{oi} + \sum_{m=1}^{NPOL} \sum_{k=1}^{NIDV} C_{mk,i} v_k^m \quad (i = 1, 2, \dots, NDDV) \quad (1)$$

where NPOL is the degree of the polynomial expression for each dependent design variable, $C_{mk,i}$ is the i th linking coefficient, C_{oi} is the i th additive constant, v_i is the i th dependent design variable, and v_k is the independent design variable. With matrix notation, equation (1) may be written

$$\{v\} = \{C\}_o + \sum_{m=1}^{NPOL} [C]_m \{v\}^m \quad (2)$$

STIFFNESS AND MASS MATRIX DERIVATIVES

The derivatives of the stiffness matrix $[K]$ and of the mass matrix $[M]$ with respect to the independent design variable v_k are needed in the subsequent calculations of sensitivity derivatives. Two methods are used for calculating these matrix derivatives: analytical and finite differences.

Analytical Method

When the stiffness or the mass matrix of a finite element is linearly related to a design variable, the analytical derivative of the matrix is obtained by simply setting the design variable to unity and calculating the corresponding matrix. When a nonlinear relationship exists between the stiffness or the mass matrix and the design variables or when design-variable linking is used, or both, the situation is more complicated and chain-rule partial differentiation is needed to formulate generalized analytical expressions. For a beam element the stiffness matrix is linearly related to four structural definition parameters (A , I_1 , I_2 , and J_0). Analytical

expressions may be written to relate the design variables to these structural definition parameters, and the derivative of the stiffness matrix can be calculated by using

$$\frac{\partial [K]}{\partial v_i} = \frac{\partial [K]}{\partial A} \frac{\partial A}{\partial v_i} + \frac{\partial [K]}{\partial I_1} \frac{\partial I_1}{\partial v_i} + \frac{\partial [K]}{\partial I_2} \frac{\partial I_2}{\partial v_i} + \frac{\partial [K]}{\partial J_o} \frac{\partial J_o}{\partial v_i} \quad (3)$$

or

$$\frac{\partial [K]}{\partial v_i} = \sum_{j=1}^{NSDP} \frac{\partial [K]}{\partial P_j} \frac{\partial P_j}{\partial v_i} \quad (4)$$

where NSDP is the number of structural definition parameters, $\partial [K]/\partial P_j$ is computed by substituting unity for P_j , and $\partial P_j/\partial v_i$ is calculated by differentiating the analytical expression relating v_i to P_j . The analogous expression for mass matrix derivatives is

$$\frac{\partial [M]}{\partial v_i} = \sum_{j=1}^{NSDP} \frac{\partial [M]}{\partial P_j} \frac{\partial P_j}{\partial v_i} \quad (5)$$

If design-variable linking is used, the independent and dependent design variables are related by equation (1) and the expressions for derivatives with respect to the independent design variables are

$$\frac{\partial [K]}{\partial v_k} = \sum_{i=1}^{NDDV} \sum_{j=1}^{NSDP} \frac{\partial [K]}{\partial P_j} \frac{\partial P_j}{\partial v_i} \frac{\partial v_i}{\partial v_k} \quad (6)$$

and

$$\frac{\partial [M]}{\partial v_k} = \sum_{i=1}^{NDDV} \sum_{j=1}^{NSDP} \frac{\partial [M]}{\partial P_j} \frac{\partial P_j}{\partial v_i} \frac{\partial v_i}{\partial v_k} \quad (7)$$

Example of Analytical Derivative of Stiffness Matrix

To crystallize the ideas and terms involved, differentiation of the stiffness matrix for a beam modeled by three E23 channel-section elements (fig. 2) is performed. The structural definition parameters consist of the cross-sectional area A and the moments of inertia I_1 , I_2 , and J_o . The dependent design variables are the dimensions of the channel B_1 , B_2 , and t . There are three independent design

variables, each associated with an element and denoted V_1 , V_2 , and V_3 . The dependent design variables are linked to the independent design variables by a 9×3 matrix $[C]$. Thus, for the structure in figure 2,

$$\{P\}^T = \{A \quad I_1 \quad I_2 \quad J_o\} \quad (8)$$

$$\{v\}^T = \begin{matrix} \{B_1 B_2 t & B_1 B_2 t & B_1 B_2 t\} \\ \text{Element 1} & \text{Element 2} & \text{Element 3} \end{matrix} \quad (9)$$

$$\{V\}^T = \{V_1 \quad V_2 \quad V_3\} \quad (10)$$

$$\{v\} = [C]\{V\} \quad (11)$$

The structural definition parameters are related to the dependent design variables by equations given in reference 4, as follows:

$$\left. \begin{aligned} P_1 &= A = (2B_1 + B_2)t \\ P_2 &= I_1 = \frac{B_1(B_2 + 2t)^3}{12} - \frac{(B_1 - t)B_2^3}{12} \\ P_3 &= I_2 = \frac{2tB_1^3}{12} + 2tB_1\left(\frac{B_1}{2} - c\right)^2 + \frac{B_2t^3}{12} + B_2t\left(c - \frac{t}{2}\right)^2 \\ P_4 &= J_o = \frac{1}{3}(2B_1 + B_2)t^3 \\ c &= \left(B_1^2t + \frac{B_2t^2}{2}\right)\frac{1}{A} \end{aligned} \right\} \quad (12)$$

From equation (6),

$$\frac{\partial [K]}{\partial v_k} = \sum_{i=1}^9 \sum_{j=1}^4 \frac{\partial [K]}{\partial P_j} \frac{\partial P_j}{\partial v_i} \frac{\partial v_i}{\partial v_k} \quad (13)$$

where

$$\frac{\partial [K]}{\partial p_j} = [K] \Big|_{p_j=1} \quad (j = 1, 2, 3, 4) \quad (14)$$

$\partial p_j / \partial v_i$ is obtained from differentiating equations (12) and noting equation (9), and $\partial v_i / \partial v_k$ is obtained from differentiating equation (11).

Finite-Difference Method

The stiffness and mass matrix derivatives can also be computed by finite differences, as follows:

$$\frac{\partial [K]}{\partial v_k} \approx \frac{\Delta [K]}{\Delta v_k} = \frac{[K]_k^n - [K]^o}{v_k^n - v_k^o} \quad (15)$$

$$\frac{\partial [M]}{\partial v_k} \approx \frac{\Delta [M]}{\Delta v_k} = \frac{[M]_k^n - [M]^o}{v_k^n - v_k^o} \quad (16)$$

where $[K]_k^n$ and $[M]_k^n$ are the perturbed stiffness and mass matrices (formed by incrementing the k th independent design variable), $[K]^o$ and $[M]^o$ are the original (unperturbed) stiffness and mass matrices, and v_k^n and v_k^o are the perturbed and original values of the k th independent design variable. Since this method operates with the independent design variables directly, it does not require looping over the dependent design variables or the structural parameters indicated by equations (6) and (7).

Finite Differences With LSK Processor

In some instances the finite-difference calculation of the stiffness matrix may be improved by use of the LSK processor (table 1). This processor selects appropriate combinations of elements (submatrices) of the global stiffness matrix $[K]$ in a sparse matrix form (called LS-format) and performs the necessary finite differencing. For this case the differencing must be with respect to the dependent design variables, as follows:

$$\frac{\partial [K]}{\partial v_i} \approx \frac{\Delta [K]}{\Delta v_i} = \frac{[K]_i^n - [K]^o}{v_i^n - v_i^o} \quad (17)$$

If design-variable linking is used (eq. (1)),

$$\frac{\partial [K]}{\partial v_k} = \sum_{i=1}^{NDDV} \frac{\partial [K]}{\partial v_i} \frac{\partial v_i}{\partial v_k} \quad (18)$$

where $\partial v_i / \partial v_k$ is obtained by differentiating equation (1). Using the LSK processor to operate on selected portions of the global stiffness matrix is more efficient than summing and multiplying stiffness matrices having the dimensions of the entire structure. Hence, depending on the ratio of the number of dependent design variables (NDDV) to the number of independent design variables (NIDV), the finite-difference methods of equation (15) or of equations (17) and (18) may be more efficient than the analytical method of equation (6). A large ratio of dependent to independent design variables and a large number of structural parameters needed to specify a particular element favor the use of one of the finite-difference methods for calculating $\partial [K] / \partial v_k$.

DISPLACEMENT DERIVATIVES

Analytical Method

In the analytical method, derivatives are computed from the governing finite-element equations. For static finite-element structural analysis, the equilibrium equation is

$$[K]\{u\} = \{f\} \quad (19)$$

where $\{u\}$ is the vector of nodal displacements and $\{f\}$ is the applied load vector. Differentiating equation (19) with respect to the independent design variable v_k gives

$$[K] \frac{\partial \{u\}}{\partial v_k} = \frac{\partial \{f\}}{\partial v_k} - \frac{\partial [K]}{\partial v_k} \{u\} = \{F\}_k \quad (20)$$

or

$$[K] \left[\frac{\partial u}{\partial v} \right] = [F] \quad (21)$$

where $\{F\}_k$ is the k th pseudo applied load vector and $[\partial u / \partial v]$ and $[F]$ are, respectively, matrices for which the columns are the displacement derivatives and pseudo load vectors which correspond to individual independent design variables. If

the applied load vector $\{f\}$ is not a function of the design variables, then the term $\partial\{f\}/\partial v_k$ is equal to zero.¹ For this case the pseudo load vector becomes

$$\{F\}_k = - \frac{\partial[K]}{\partial v_k} \{u\} \quad (22)$$

The analytical method consists of solving equation (20) for $\partial\{u\}/\partial v_k$ using analytically computed derivatives for $\partial[K]/\partial v_k$ (eq. (6)). Equation (20) is solved by the same solution algorithm used for solving equation (19), taking advantage of the fact that the factored form of $[K]$ is available from the solution of equation (19). Substitution of equation (6) into equation (20) leads to

$$[K] \frac{\partial\{u\}}{\partial v_k} = \frac{\partial\{f\}}{\partial v_k} - \sum_{i=1}^{NDDV} \sum_{j=1}^{NSDP} \frac{\partial[K]}{\partial p_j} \frac{\partial p_j}{\partial v_i} \frac{\partial v_i}{\partial v_k} \{u\} \quad (23)$$

Semianalytical Method

The semianalytical method contains the analytical expression for the displacement derivatives (eq. (20)) with finite-difference derivatives of the stiffness matrix (eq. (15) or eqs. (17) and (18)). If equations (15) and (20) are used, the following expression results:

$$[K] \frac{\partial\{u\}}{\partial v_k} = \frac{\partial\{f\}}{\partial v_k} - \left(\frac{[K]_k^n - [K]^o}{v_k^n - v_k^o} \right) \{u\} = \{F\}_k \quad (24)$$

where $\partial\{f\}/\partial v_k$ can be calculated analytically or by finite differences and the term in parentheses may be obtained by finite differences from equation (15) or by the LSK processor (eqs. (17) and (18)). Derivatives are calculated by creating individual pseudo load vectors $\{F\}_k$ and solving equation (24) using the factored $[K]$ matrix from equation (19). This formulation is a significant simplification of the analytical approach of reference 4 in that it avoids the need for element-dependent manipulations to handle bending-type elements. It also eliminates the complications in the analytical method when design-variable linking is included (eqs. (1) and (23)).

Finite-Difference Method

The simplest method to implement, but the most time consuming computationally (especially for large finite-element models), is the finite-difference method. In addition, the accuracy of the finite-difference method depends upon the perturbation

¹One practical case when this term is not zero is when thermal loads are included.

step size. In this method, the original structure is analyzed; the structure is then modified by perturbing a design variable and reanalyzed. The displacement derivative is

$$\frac{\partial \{u\}}{\partial v_k} \approx \frac{\{u\}_k^n - \{u\}^o}{v_k^n - v_k^o} \quad (25)$$

where $\{u\}_k^n$ is the displacement vector due to a perturbation in the kth independent design variable and $\{u\}^o$ is the displacement vector of the original structure.

STRESS DERIVATIVES

Analytical Method

Element stresses $\{\sigma\}$ are related to joint displacements and element temperatures by the equation

$$\{\sigma\} = [S]\{u\} - [G]\{\alpha\}(T - T_o) \quad (26)$$

where $[S]$ is the stress-displacement matrix, $[G]$ is the stress-temperature matrix, $\{\alpha\}$ is the vector of thermal expansion coefficients, T_o is the stress-free element temperature, and T is the actual element temperature. Upon differentiation with respect to the kth independent design variable, the general expression for stress derivatives is

$$\frac{\partial \{\sigma\}}{\partial v_k} = [S] \frac{\partial \{u\}}{\partial v_k} + \sum_{i=1}^{NDDV} \sum_{j=1}^{NSDP} \frac{\partial [S]}{\partial P_j} \frac{\partial P_j}{\partial v_i} \frac{\partial v_i}{\partial v_k} \{u\} \quad (27)$$

Equation (27) neglects any dependence of temperature on the design variables.

For rods (E23), membranes (E31 and E41), and shear panels (E44), the matrix $[S]$ is independent of the section-property design variables; thus, the second term on the right side of equation (27) is zero. For bending-type elements, beams (E21) and plates (E32, E33, E42, and E43), the stress-displacement matrices are dependent on the element cross-sectional geometry, and this dependence must be included in the stress derivative calculations. This was done in reference 4 for the channel-section beam in figure 2 by using specific, user-defined subroutines.

Semianalytical Method

In this method (reported in ref. 11), the perturbed displacement vector $\{u\}_k^n$ is approximated by a truncated Taylor series expansion as follows:

$$\{u\}_k^n = \{u\}^o + \frac{\partial \{u\}}{\partial v_k} (v_k^n - v_k^o) \quad (28)$$

where $\partial\{u\}/\partial v_k$ is calculated with equation (24). The perturbed stress vector is given by

$$\{\sigma\}_k^n = [S]_k^n \{u\}_k^n - [G]\{\alpha\}[G]\{\alpha\}(T - T_o) \quad (29)$$

and the derivative is

$$\frac{\partial\{\sigma\}}{\partial v_k} \approx \frac{\{\sigma\}_k^n - \{\sigma\}^o}{v_k^n - v_k^o} \quad (30)$$

where

$$\{\sigma\}^o = [S]\{u\}^o - [G]\{\alpha\}(T - T_o) \quad (31)$$

Finite-Difference Method

The stress derivatives are calculated in a manner analogous to the displacement derivatives. The structure is modified and reanalyzed and the derivatives are approximated as

$$\frac{\partial\{\sigma\}}{\partial v_k} \approx \frac{\{\sigma\}_k^n - \{\sigma\}^o}{v_k^n - v_k^o} \quad (32)$$

where $\{\sigma\}_k^n$ is the stress vector due to a perturbation in the kth independent design variable calculated with equation (29) and $\{\sigma\}^o$ is the original stress vector.

DERIVATIVES OF VIBRATION AND BUCKLING EIGENVALUES AND EIGENVECTORS

Methodologies for calculating derivatives of eigenvalues and eigenvectors have been presented in references 4, 6, 8, and 9. The methods which have been implemented in EAL are an analytical method (ref. 8) and the finite-difference method.

Analytical Method

This method is implemented in EAL using runstreams based on the development in reference 4 and the theory in reference 8. The matrix equation for free vibrations is

$$\left([K] - \omega_j^2 [M]\right)\{\phi\}_j = 0 \quad (33)$$

where $\{\phi\}_j$ and ω_j are, respectively, the vector of the mode shape and the scalar frequency corresponding to vibration mode j , $[M]$ is the mass matrix, and $[K]$ is the stiffness matrix. Normalizing the modes with respect to the mass matrix gives

$$\{\phi\}^T [M] \{\phi\} = [I] \quad (34)$$

where $[I]$ is the identity matrix. Differentiating equation (33) with respect to V_k gives

$$\left([K] - \omega_j^2 [M] \right) \frac{\partial \{\phi\}_j}{\partial V_k} = \frac{\partial \omega_j^2}{\partial V_k} [M] \{\phi\}_j - \frac{\partial [K]}{\partial V_k} \{\phi\}_j + \omega_j^2 \frac{\partial [M]}{\partial V_k} \{\phi\}_j \quad (35)$$

Premultiplying equation (35) by $\{\phi\}^T$ and using equations (33) and (34) gives

$$\frac{\partial \omega_j^2}{\partial V_k} = \{\phi\}_j^T \frac{\partial [K]}{\partial V_k} \{\phi\}_j - \omega_j^2 \{\phi\}_j^T \frac{\partial [M]}{\partial V_k} \{\phi\}_j \quad (36)$$

Since $[K] - \omega_j^2 [M]$ is singular, a direct solution of equation (35) is not attempted. The value of one component of $\partial \{\phi\}_j / \partial V_k$ is fixed, yielding a particular solution $\{Q\}_j$ to equation (35). This is accomplished in EAL by identifying the component of the eigenvector with the largest absolute value and constraining to zero the corresponding component of the eigenvector derivative. The eigenvector $\{\phi\}_j$ is the complementary solution to equation (35). Thus, the expression for the eigenvector derivative is

$$\frac{\partial \{\phi\}_j}{\partial V_k} = \{Q\}_j + C \{\phi\}_j \quad (37)$$

The value of the multiplier C is obtained by substituting the expression for $\partial \{\phi\}_j / \partial V_k$ from equation (37) into the following:

$$2 \{\phi\}_j^T [M] \frac{\partial \{\phi\}_j}{\partial V_k} = - \{\phi\}_j^T \frac{\partial [M]}{\partial V_k} \{\phi\}_j \quad (38)$$

Equation (38) is the derivative of the expression for the normalization of modes with respect to the mass matrix (eq. (34)). The resultant equation for C is

$$C = - \{\phi\}_j^T [M] \{Q\}_j - \frac{1}{2} \{\phi\}_j^T \frac{\partial [M]}{\partial V_k} \{\phi\}_j \quad (39)$$

For buckling eigenvalue and eigenvector derivatives, the preceding equations apply directly, with the buckling mode shape $\{\Phi\}_j$ and load λ_j substituted for the vibration mode shape $\{\phi\}_j$ and frequency ω_j^2 and with the negative of the geometric stiffness matrix $-[K_g]$ substituted for the matrix $[M]$. In the present implementation, derivatives of the matrices $[K]$, $[M]$, and $[K_g]$ were calculated by finite differences.

Finite-Difference Method

The equations for finite-difference derivatives of eigenvalues and eigenvectors are analogous to those previously shown for displacement and stress derivatives (eqs. (25) and (32)). When the finite-difference method is used to obtain derivatives of eigenvectors, careful attention must be paid to accuracy of the eigenvectors. Problems arise because eigenvalue routines generally base convergence on accuracy of the eigenvalues, whereas the associated eigenvectors may not be as accurate. Hence, when differencing the eigenvectors, further errors are introduced which may make the derivatives inaccurate.

IMPLEMENTATION OF DERIVATIVE CAPABILITY IN EAL

An outline of the implementation of the derivative capability in EAL is given in the appendix. For brevity, only the implementation of the semianalytical method is discussed in the appendix, which includes descriptions of the overall procedure, descriptions of the key portions of the EAL runstreams, and a listing of the input for the problem of the box beam.

NUMERICAL EXAMPLES AND RESULTS

Four example problems were investigated to verify the derivative implementation and to compare the methods. The problems were a swept wing, a box beam, a stiffened cylinder with a cutout, and a space radiometer-antenna truss. Comparisons of solution times for the various derivative calculation methods are given, and a convergence study was performed for the wing problem to verify the accuracy of the finite-difference results and to establish an appropriate perturbation in the design variables for the remaining finite-difference calculations. Analytical derivatives (when available) were used to assess the accuracy of the other methods. All numerical calculations were performed on the CDC® CYBER 175 computer under NOS 1.4.

Swept Wing

Optimization of the swept wing shown in figure 3 has been investigated and reported by several researchers (refs. 17 to 19). This example was a moderately complex representation of a wing modeled by rods (E23), triangular membranes (E31), and shear panels (E44). Since bending elements were not used, the calculation of derivatives with the analytical method was straightforward, and the results were used to check the accuracy of the semianalytical and finite-difference methods. The geometry and node numbering are shown in figure 3; nodal coordinates are listed in table 3 and design variables are described in table 4. In this problem, 32 design variables controlled the section properties of 150 elements. The wing was aluminum, had a constrained root, and was subjected to two load conditions. (See table 5.) Load condition 1 was approximately equivalent to a uniform pressure loading of

0.556 psi. Load condition 2 had the same total load, but the distribution was changed to move the center of pressure forward. The wing was symmetric with respect to the x-y plane (u and v displacements were zero on the plane $z = 0$), and hence only half the wing was modeled. The model consisted of triangular membrane elements for the skin, rod elements for the spar caps, and shear panels for the rib and spar webs. (See fig. 4.) Forty-four transverse rod elements (not shown) were added at the vertical edges of the shear panels to provide the necessary stiffness throughout the depth of the wing. The thicknesses t_i were 0.2 in. for E31 elements 1 to 24, 0.1 in. for E31 elements 25 to 60, and 0.2 in. for all E44 elements. The cross-sectional area A_i was 0.02 in^2 for E23 elements 1 to 20 and 0.2 in^2 for E23 elements 21 to 64.

A study was performed to determine the perturbation step size needed to obtain sufficiently accurate results for the finite-difference and semianalytical methods. The results are shown in table 6(a). The value of the largest displacement derivative value ($\partial w / \partial V_{14}$ at joint number 41 for load condition 1) was used to judge convergence. As indicated in table 6(a), the finite-difference result, with a 1-percent increment in the design variable, was within 1 percent of the analytical results. Use of perturbations smaller than 0.01 percent led to degradation of accuracy. The semianalytical method was capable of duplicating the analytical result with a 1-percent perturbation in the design variable. Although the acceptable perturbation sizes are generally problem dependent, this dependence was not significant. In all subsequent calculations, perturbations of 1 percent were used.

Comparisons of solution times for displacement derivatives are also presented in table 6(b). The finite-difference method was slowest (700 sec), the semianalytical method required 419 sec without the LSK processor and 161 sec with the LSK processor, and the analytical method required 135 sec. Hence, the use of the LSK processor makes the semianalytical method significantly more effective and makes it competitive with the analytical method.

Box Beam

The next problem was a cantilevered box beam (fig. 5). This example was used to verify displacement derivative runstreams for all the EAL structural elements listed in table 2.

The geometry and joint numbering are shown in figure 5, and elements are illustrated in figure 6. The thickness of all two-dimensional elements was 0.1 in. The cross-sectional area of the rod elements was 1.0 in^2 , and the beam elements had tube sections with an inner radius of 2.0 in. and an outer radius of 2.5 in. The material was aluminum with the following material properties: $\rho = 0.096 \text{ lb/in}^3$, $E = 10.6 \times 10^6 \text{ psi}$, and $\nu = 0.3$. The applied load was composed of 10 000-lb forces in the positive z direction at nodes 27 and 28. The cross-sectional area or thickness of each group of elements was considered to be a separate design variable. The problem had 10 dependent design variables and 9 independent design variables. The inner and outer radii of the tube elements were linked to an independent variable r by the equation

$$\begin{Bmatrix} r_i \\ r_o \end{Bmatrix} = \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} r + \begin{Bmatrix} 0 \\ 0.5 \end{Bmatrix} \quad (40)$$

Displacement derivatives with respect to each independent design variable were calculated with the finite-difference method and the semianalytical method.² Results shown in table 7(a) indicate run times of 133 sec for the finite-difference method and 63 sec for the semianalytical method with the LSK processor. The reduction in solution time for the semianalytical method with LSK over the finite-difference method was smaller for this example than for the swept wing because the swept wing problem had more independent design variables (32 instead of 9). Values shown in table 7(b) reveal which design variables would be most effective to increment to reduce tip deflections. As expected, the upper and lower covers (coupled plate elements E43 and E33) had the greatest effect on limiting tip deflections. In addition, since the E32 and E42 elements had only bending stiffness and the bending loads in the covers were negligible, the derivatives with respect to thickness of these elements were negligible. The results in table 7(b) suggest that derivatives can be used directly to guide improvements in a structural design apart from their use in formal optimization. In some instances substantial improvements may be realized (ref. 20).

Stiffened Cylinder With Cutout

Optimization studies of a cantilevered stiffened cylinder with a cutout (fig. 7) were reported in references 4 and 21. Node point locations are given in table 8. Element types and locations are shown in figure 7(b). The cylinder model had 352 degrees of freedom and was stiffened by 5 equally spaced rings along the length and 16 equally spaced stringers around the circumference. Rotations about the Y-axis at all node points and all translations at node points 1 to 16 were constrained to zero. The rings and stringers were modeled by beam (E21 channel sections) and rod elements (E23). Rectangular panels between rings and stringers were modeled by membrane elements (E41). The cross-sectional area of all rod elements was 0.646 in², the thickness of the rectangular membranes was 0.0394 in., and the channel-section dimensions are shown in figure 7(b). The material was aluminum alloy with $E = 10.8 \times 10^6$ psi and $\nu = 0.3$. The independent design variables for this problem included the area of the rods, the thickness of the membrane elements, and a scale factor which controlled the cross-sectional dimensions of the beam elements. The loading consisted of two equal and opposite concentrated axial forces of 20 000 lb applied at the free end.

Derivatives of displacements, stresses, vibration frequencies and mode shapes, and buckling loads and mode shapes were calculated. Computer times for calculating derivatives are presented in table 9. Use of the semianalytical method without the LSK processor for displacement derivatives resulted in a reduction in solution time of about 25 percent over the finite-difference method. Using the semianalytical method with the LSK processor resulted in a reduction in solution time of 28 percent over the finite-difference method. The reason solution times for the semianalytical methods were so similar for this case was that the ratio of the number of independent to the number of dependent design variables was only 3 to 5. As this ratio decreases, the advantage of the LSK processor increases. (In the problem of the swept wing, the ratio was 32 to 150 and the LSK processor had a much larger effect.)

Stress derivatives were calculated by finite differences (122 sec) and semi-analytically (95 sec). The savings in solution time for finite differences relative

²Derivatives for the membrane and rod elements were computed with the analytical method.

to the semianalytical method were not as dramatic in this problem as in the problems of the swept wing or the box beam because of the smaller number of design variables.

Computer times for calculating derivatives of vibration frequencies, vibration mode shapes, buckling loads, and buckling mode shapes with the finite-difference and analytical methods are also shown in table 9. For the vibration problem, the analytical method was 57 percent faster than finite differences, and for the buckling problem it was 37 percent faster.

Antenna Truss

A finite-element model of a 180-ft radiometer-antenna reflector is shown in figure 8. The reflector, described in reference 22, was made up of tetrahedral truss modules, and the model consisted of 109 structural node points (table 10) and 420 rod elements. The structure was composed of graphite-epoxy composite with an effective modulus of elasticity of 10.6×10^6 psi and a coefficient of thermal expansion of 0.13×10^{-6} per degree Fahrenheit. The antenna was subjected to thermal loading corresponding to Earth orbit at an altitude of 216 miles in an Earth-facing orientation. The thermal loading consisted of a combination of solar, Earth-reflected (albedo), and Earth-emitted heat flux. A transient thermal analysis of the structure for a complete orbit was performed (ref. 23) and node point temperature differences from a worst-case condition (largest temperature gradients, table 11) were used for the present calculations. Three design variables were used for this problem. They were the cross-sectional area of the elements in the upper surface (0.2530 in^2), the area of the diagonal elements joining the upper and lower surfaces (0.1741 in^2), and the area of the elements in lower surface (0.2530 in^2). Both stress and displacement derivatives were calculated, and for this problem the applied load vector $\{f\}$ was a function of the design variables. (See eq. (20).)

Solution times for displacement and stress derivatives are summarized in table 12(a). The semianalytical method was 39 percent faster than the finite-difference method for displacement derivatives and 29 percent faster for stress derivatives for this problem. As indicated by the tabulated derivatives in table 12(b), the most effective way to reduce the center deflection of the reflector would be to decrease the areas of the upper-surface elements.

CONCLUDING REMARKS

A capability for computing structural-sensitivity derivatives has been implemented in the general purpose, finite-element computer program denoted the Engineering Analysis Language (EAL) System. This paper presented the development of runstreams which calculate derivatives of displacements, stresses, vibration frequencies and mode shapes, and buckling loads and mode shapes with respect to structural variables which include thicknesses, areas, and moments of inertia. Linear and nonlinear design-variable linking representations were included. Three methods for computing the derivatives were documented: analytical (which calculates exact structural-sensitivity derivatives from the governing equations), semianalytical (which combines finite-difference derivatives of mass, stiffness, and geometric stiffness matrices with analytical expressions for derivatives), and finite differences. The derivative capability was demonstrated for the following four structures: a swept wing, a box beam, a stiffened cylinder with a cutout, and a space radiometer-antenna truss. Comparisons of solution times for the various methods were described. A convergence study was performed for the swept wing to

verify the accuracy of the finite-difference results and to establish an appropriate increment in the design variables for use in the subsequent finite-difference calculations. Analytical derivatives of displacements for the wing and of vibration frequencies of the cylinder served as benchmarks to assess the accuracy and efficiency of the finite-difference and semianalytical methods. Results indicated that the semianalytical method for calculating displacement and stress derivatives was efficient, general, and straightforward to implement. Further, the semianalytical method was not adversely affected by the presence of bending elements or design-variable linking.

Langley Research Center
National Aeronautics and Space Administration
Hampton, VA 23665
February 8, 1984

APPENDIX

SYSTEM OF EAL RUNSTREAMS FOR CALCULATING STRUCTURAL-SENSITIVITY DERIVATIVES WITH SEMIANALYTICAL METHOD

Overall Procedure

This appendix describes a system of EAL runstreams used to calculate structural-sensitivity derivatives with the semianalytical method. A flowchart for the overall system is shown in figure 9. Runstreams or groups of runstreams presented at the end of this appendix are keyed by numerals to specific portions of flowcharts shown in figures 10 to 12. The semianalytical method uses equations (15) and (16) to calculate derivatives of the stiffness and mass matrices and uses equations (24) and (28) to (30) to calculate the displacement and stress derivatives.

As shown in figure 10, derivatives of the stiffness, mass, and geometric stiffness matrices are calculated by finite differences. The procedure is as follows: loop over the independent design variables, perturb a design variable, link design variables to form vector of dependent design variables, loop over the dependent design variables and modify the structure, calculate new stiffness and mass matrices and geometric stiffness matrix (if appropriate), differentiate the resulting matrices, and restore the original value of independent design variable V_k . The numerals beside specific boxes correspond to runstreams described subsequently.

Displacement derivatives (fig. 11) are calculated by looping over the independent design variables, building a pseudo load matrix (eq. (20)) and solving equation (24). The acquired displacement derivatives are then used to approximate a perturbed displacement vector $\{u\}_k^n$ (eq. (28)) and then to approximate a perturbed stress vector (eq. (29)). The stress derivatives are then calculated by finite differences (eq. (30)).

The flowchart which illustrates the calculation of derivatives of vibration and buckling eigenvalues is shown in figure 12. The outer loop is on the mode for which derivatives are required. The two inner loops are both over the independent design variables. The first inner loop calculates the derivatives of the vibration frequencies and buckling loads with equation (36) and then builds a pseudo load vector (right-hand side of eq. (35)). A particular solution of equation (35) is calculated and used in the second inner loop to calculate derivatives of the eigenvectors with equations (37) and (39).

Descriptions of the key runstreams used in the system of figures 9 to 12 (particularly those which are problem dependent) are given in the following table:

APPENDIX

DESCRIPTION OF KEY RUNSTREAMS

Runstream name	Identification no. in figs. 9 to 12	Description
DRIV GRD4	1	Driver runstream which calculates derivatives semianalytically; the entire system can be run by the statement *DCALL (DRIV GRD4)
INIT MODL	2	Sequence of EAL processors (TAB, ELD, TAN, and AUS) which describes the initial structural model and load cases; the example shown in the runstream listing is for the box beam
SET PARA	3	<p>Initializes following parameters used by system to determine specifics of derivative information required:</p> <p>NLST Number of load sets</p> <p>MODT Number of modes for which derivatives are computed by method of reference 8</p> <p>MODE Same as MODT</p> <p>OLIB Output library for intermediate buckling and vibration derivatives</p> <p>OTIB Output library for intermediate stress or displacement derivatives</p> <p>NPOL Degree of polynomial used in linking design variables (eq. (1))</p> <p>DISP If equal to 1, displacement derivatives are calculated</p> <p>STRD If equal to 1, stress derivatives are calculated</p> <p>BUCD If equal to 1, buckling load and mode shape derivatives are calculated; if equal to 2, only buckling load derivatives are calculated</p> <p>VIBD Same as BUCD for vibration derivative</p> <p>NJ Number of joints in model</p> <p>NDV Number of independent design variables</p> <p>NDVD Number of dependent design variables</p> <p>NDVA Number of rod and membrane dependent design variables (E23, E31, E41, E44)</p> <p>NEDV Number of beam dependent design variables (E21)</p> <p>NPDV Number of plate dependent design variables (E32, E33, E42, E43)</p> <p>NDVX Number of shape-dependent design variables</p> <p>DR Incrementing factor for design variables for finite-difference calculations</p> <p>DE21, DE23, ..., DE44 } Number of dependent design variables for each element</p> <p>NE21, NE23, ..., NE44 } Number of independent design variables for each element</p> <p>MODE NUM TABLE Name of table of mode numbers for which derivatives are taken</p>

APPENDIX

Runstream name	Identification no. in figs. 9 to 12	Description
DESV DFN	4	Set up tables denoted DVA DFN, DVP DFN, and DVB DFN which define membrane, plate, and beam dependent design variables, respectively; element types are defined by a numeral: E23-1, E31-2, E32-3, E33-4, E41-5, E42-6, E43-7, E44-8, and E21-9; the type of beam used in EAL in section 3.1.9 of reference 12 is denoted by a code as follows: TUBE - 1, BOX - 2, TEE - 3, ANG - 4, WFL - 5, CHN - 6, ZEE - 7, GIVN - 8, DSY - 9; beam design variable numbers are designated by the following codes: $b_1 - 1$, $t_1 - 2$, $b_2 - 3$, $t_2 - 4$, $b_3 - 5$, $t_3 - 6$, $I_1 - 7$, $\alpha_1 - 8$, $I_2 - 9$, $\alpha_2 - 10$, $a - 11$, $f - 12$, $f_1 - 13$, $Z_1 - 14$, $Z_2 - 15$, $\theta - 16$, $q_1 - 17$, $q_2 - 18$, $q_3 - 19$; each of the variables shown is described in section 3.1.9 of reference 12; the order in which the dependent design variables are listed (starting with the membrane, then plate, and then beam design variables) are the actual locations of each design variable in the dependent design variable vector (see INTL DESV); the only restriction is that the structural parameters of beams with the same section property number must be listed together in the DVB DFN table; this permits the linking of several section properties of a beam; this could be used to facilitate the scaling of an entire beam cross section by one design variable or scale factor
INTL DESV	5	Initializes design variables by forming a table of initial design variables called DESV CNMN whose only limitation is that when linked it will produce the same structural parameters as the initial structural model (see INIT MODL)
COEF LINK	6	Series of tables which define the coefficients for nonlinear design-variable linking as expressed by equation (1) and rewritten as $\{v\} = \{C_0\} + [C_1]\{v\} + [C_2]\{v^2\}, \dots, + [C_M]\{v^M\}$ where $\{C_0\}$ is represented as table COEF DV 1 0, $[C_1]$ is COEF DV 1 1, $[C_2]$ is COEF DV 1 2, ..., $[C_M]$ is COEF DV 1 M; each block of the above tables represents a column in the $[C]$ matrices; this runstream must be included in the system of runstreams even if linking is not used

APPENDIX

Runstream Listings

This section of the appendix contains a listing of the EAL runstreams used to calculate derivatives of displacement, stress, and vibration and buckling eigenvalues and eigenvectors with the semianalytical method.

```
* CHAR $*!";Z
$*ECHU 1,2,3,4
*ABORT 1
*XQT U1
$
$----- 1
$
*(DRIV GRD4)          ENDDG4
$DRIVER PROGRAM TO CALCULATE DU/DCAPV BY THE SEMI-ANALYTICAL METHOD
*DCALL(INIT MODL)
*XQT DCU
DUPLICATE 1 2
DUPLICATE 1 3
DUPLICATE 1 4
DUPLICATE 1 5
*DCALL(SET PARA)
*DCALL(DESX DFN)
*DCALL(INTL DESV)
*DCALL(COEF LINK)
*DCALL(CALC GRD4)
*DCALL(PRT SUB)
*RETURN
*                      ENDDG4
$
$----- 2
$
*(INIT MODL)          ENDINM
$$          $$          INITIAL MODEL STRUCTURAL DEFINITION
*XQT TAB
START 28
TITLE BOX BEAM GRADIENT CALCULATION
TEXT
'GRADIENT CALCULATION OF A BOX-BEAM WHICH USES
'E21,E23,E31,E32,E33,E41,E42,E43,AND E44 ELEMENTS
$INPUT JOINT LOCATIONS
JLOC
1 0. 0. 0. 0. 10. 0. 2 1 7
2 60. 0. 0. 60. 10. 0.
15 0. 0. 10. 0. 10. 10. 2 1 7
2 60. 0. 10. 60. 10. 10.
$SPECIFY MATERIAL PROPERTIES
MATERIAL CONSTANTS
1 10.6E+6 .3 .096
$DESCRIBE MATERIAL REFERENCE FRAME
MREF
FORMAT=2
1,1,1000.,0.,1000.
2,1,1000.,10.,1000.
$BEAM ELEMENT SECTION PROPERTIES
BA
TUBE 1, .25,.3
$ROD ELEMENT AREAS
dC
1,1.
$MEMBRANE AND PLATE ELEMENT THICKNESSES
SA
NMAT=1
1,.1$E31 ELEMENTS
2,.1$E32 ELEMENTS
3,.1$E33 ELEMENTS
4,.1$E41 ELEMENTS
5,.1$E42 ELEMENTS
6,.1$E43 ELEMENTS
$SHEAR PANEL THICKNESSES
SB
1,.1
$CONSTRAINT DEFINITION
```

APPENDIX

```

CON=1
ZERO 1,2,3,4,5,6;1,2
ZERO 1,2,4,5,6;15,16
*XQT ELD
$BEAM ELEMENT DEFINITION
E21
NMAT=1
NSECT=1
NREF=1
1 3 1 6
15 17 1 6
NREF=2
2 4 1 6
16 18 1 6
$ROD ELEMENT DEFINITION
E23
NMAT=1
NSECT=1
1 15 1 1 7 2
2 16 1 1 7 2
$TRIANGULAR MEMBRANE ELEMENT DEFINITION
E31
NMAT=1
NSECT=1
8 10 7
7 9 10
9 11 10
10 12 11
12 14 11
11 13 14
$UNCOUPLD TRIANGULAR PLATE ELEMENT DEFINITION
E32
NMAT=1
NSECT=2
8 10 7
7 9 10
9 11 10
10 12 11
12 14 11
11 13 14
$COUPLED TRIANGULAR PLATE ELEMENT DEFINITION
E33
NMAT=1
NSECT=3
1 3 2
2 4 3
4 6 3
3 5 6
5 7 6
6 8 7
$MQUADRILATERAL MEMBRANE ELEMENT DEFINITION
E41
NMAT=1
NSECT=4
21 23 24 22 1 3 1
$UNCOUPLD QUADRILATERAL PLATE ELEMENT DEFINITION
E42
NMAT=1
NSECT=5
21 23 24 22 1 3 1
$COUPLED QUADRILATERAL PLATE ELEMENT DEFINITION
E43
NMAT=1
NSECT=6
15 17 18 16 1 3 1
$SHEAR PANEL DEFINITION
E44
NMAT=1
NSECT=1
1 3 17 15 1 6 1
2 4 18 16 1 6 1
*XQT TAN
$APPLIED LOADS DEFINITION
*XQT AUS
SYSVEC;APPLIED FORCE 1
I=3;J=27,28; 10000. 10000.
*RETURN

```

APPENDIX

```

*                               ENDINM
$
$----- 3
$
*(SET PARA)                      ENDPAR
$$      $$ SET CONTROL PARAMETERS
INLST=1$NUMBER OF LOAD SETS
ILCAS=1$NUMBER OF LOAD CASES
IMJDT=2$TOTAL NUMBER OF MODES
IOLIB=15$OUTPUT LIBRARY FOR INTERMEDIATE DYNAMIC DERIVATIVES
IGTIB=15$OUTPUT LIBRARY FOR INTERMEDIATE STRESS OR DISP. DERIVATIVES
IOLIB=15
IMODE=2
INPOL=1$DEGREE OF POLYNOMIAL LINKING
IDISP=1$DISPLACEMENT DERIVATIVES
ISTRD=1$STRESS DERIVATIVES
IBUCD=1$BUCK DERIVS.(LOAD&FREQ)
IVIBD=1$VIB. DERIVS.(LOAD&FREQ)
INJ=28$NO. OF JOINTS
INDV=9$NO. OF INDEPENDENT DESIGN VARIABLES
INDV=NDV
INDVX=0$NO. OF DEPENDENT SHAPE VARIABLES TO BE LINKED
INDVA=4 $NO. OF DEPENDENT MEMBRANE -TYPE DESIGN VARIABLES TO BE LINKED
INLNK=10$ NO. OF LINKED DESIGN VARIABLES
INDVI="NDV"
INBDV=2$ NO. OF BEAM DEPENDENT DESIGN VARIABLES TO BE LINKED
INPDV=4$ NO. OF PLATE DEPENDENT DESIGN VARIABLES TO BE LINKED
INDVD=NDVX+NDVA+NBDV+NPDV
INDVI=NDV
IDR=.001$ FACTOR FOR INCREMENTING DESIGN VARIABLE
IDE23=1$NO. OF E23 D.V.'S
IDE31=1$ NO. OF E31 D.V.'S
IDE32=1$ NO. OF E32 D.V.'S
IDE33=1$ NO. OF E33 D.V.'S
IDE41=1$ NO. OF E41 D.V.'S
IDE42=1$ NO. OF E42 D.V.'S
IDE43=1$ NO. OF E43 D.V.'S
IDE44=1$ NO. OF E44 D.V.'S
IDE21=2$ NO. OF E21 D.V.'S
INE21=1
!INX1=DS,4,1,1(1,JDF1,BTAB,1,8)
!INX2=DS,5,1,1(1,JDF1,BTAB,1,8)
!INX3=DS,6,1,1(1,JDF1,BTAB,1,8)
!INX4=DS,7,1,1(1,JDF1,BTAB,1,8)
!INX5=DS,8,1,1(1,JDF1,BTAB,1,8)
!INX6=DS,9,1,1(1,JDF1,BTAB,1,8)
$
$TABLE OF MODE NUMBERS
$
*XQT AUS
TABLE(RMODE=2,TYPE=0,NI=1,NJ="MODE");MODE NUM
J=1;1
J=2;2
$
$TABLE OF UNCONSTRAINED DEGREES OF FREEDOM
$
TABLE(RMODE=2,TYPE=0,NI=1,NJ=6);IN EX
J=1;"INX1"
J=2;"INX2"
J=3;"INX3"
J=4;"INX4"
J=5;"INX5"
J=6;"INX6"
!INX1=FREE()
!INX2=FREE()
!INX3=FREE()
!INX4=FREE()
!INX5=FREE()
!INX6=FREE()
*RETURN
*                               ENDPAR
$
$----- 4
$
*(DESV DFN)                      ENDDVD
$$      $$ DEPENDENT DESIGN VARIABLE DEFINITION
*XQT AUS

```

APPENDIX

\$MEMBRANE-TYPE DESIGN VARIABLES

TABLE(RMODE=2,TYPE=0,NI=4,NJ="NDVA");DVA DFN				
J=1,"NDVA"	ELEMENT TYPE	GROUP NO.	NO. OF ELEM.	SECT. NO.
	1	1	12	1
	2	1	6	1
	5	1	3	4
	8	1	12	1

\$PLATE-TYPE DESIGN VARIABLES

TABLE(RMODE=2,TYPE=0,NI=4,NJ="NPDV");DVP DFN				
J=1,"NPDV"	ELEMENT TYPE	GROUP NO.	NO. OF ELEM.	SECT. NO.
	3	1	6	2
	4	1	6	3
	6	1	3	5
	7	1	3	6

\$BEAM-TYPE DESIGN VARIABLES

TABLE(RMODE=2,TYPE=0,NI=5,NJ="NBDV");DVB DFN					
J=1,"NBDV"	ELEM. TYPE	GROUP NO.	NO. OF ELM.	SECT. NO.	D.V. NO.
	1	1	24	1	1
	1	1	24	1	2

*RETURN

* ENDDVD

\$----- 5

\$(INTL DESV) ENDIND

\$\$ INITIALIZE DESIGN VARIABLES

*XQT AUS

TABLE(NJ="NDV"); DESV CNMN\$DATA SET CONTAINING INITIAL VALUE OF DESIGN VARIABLES

J=1;1.

J=2;8;.1

J=9;.25

*RETURN

* ENDIND

\$----- 6

\$(COEF LINK) ENDCLK

\$\$ MATRIX OF COEFFICIENTS FOR LINKING

*XQT AUS

TABLE(NJ="NDVD");COEF DV 1 1

BLOCK 1

I=1;J=1;1.

BLOCK 2

I=1;J=2;1.

BLOCK 3

I=1;J=3;1.

BLOCK 4

I=1;J=4;1.

BLOCK 5

I=1;J=5;1.

BLOCK 6

I=1;J=6;1.

BLOCK 7

I=1;J=7;1.

BLOCK 8

I=1;J=8;1.

BLOCK 9

I=1;J=9;1.

I=1;J=10;1.

TABLE(NJ="NDVD");COEF DV 1 0

J=1,9;0.

J=10;.05

*RETURN

* ENDCLK

\$----- 7

\$(CALC GRD4) ENDGD4

\$\$ CALCULATE DERIVATIVES OF U W.R.T. DESIGN VARIABLES USING THE SEMI-ANALYTICAL METHOD

*DCALL(CALC KM)

*DCALL(DKDV DMDV)

*JNZ(DISP,262)

*JNZ(STRD,262)

*JUMP 265

*LABEL 262

*DCALL(CALC DUDV)

*LABEL 264

*JZ(STRD,265)\$TEST TO SEE IF STRESS DERIVATIVES ARE REQUIRED

APPENDIX

```

*DCALL(STRS DEKV)$CALCULATE STRESS DERIVATIVES
*LABEL 265
$TEST TO SEE IF BUCKLING DERIVATIVES ARE REQUIRED
*JZ(BUGD,266)
IN=BUGD
*JNZ,-1(N,267)
INAM9='BUCK
INAM7='BKVL
INAM8='BKMD
*DCALL(NELS METH)
*JUMP 266
*LABEL 267
*JNZ,-1(N,269)
INAM9='BUCK
INAM7='BKVL
INAM8='BKMD
*DCALL(NELS MET2)
*LABEL 269
*JNZ,-1(N,273)
INAM9='BUCK
INAM7='BKVL
INAM8='BKMD
*DCALL(MDDL METH)
*JUMP 266
*LABEL 273
*JNZ,-1(N,285)
INAM9='BUCK
INAM7='BKVL
INAM8='BKMD
*DCALL(NELS METH)
*JUMP 266
*LABEL 285
*XQT EXIT
*LABEL 266
*JZ(VIBD,268)$TEST TO SEE IF VIBRATION DERIVATIVES ARE REQUIRED
IN=VIBD
*JNZ,-1(N,270)
INAM9='VIBR
INAM7='VBVL
INAM8='VBMD
*DCALL(NELS METH)
*JUMP 268
*LABEL 270
*JNZ,-1(N,271)
INAM9='VIBR
INAM7='VBVL
INAM8='VBMD
*DCALL(NELS MET2)
*JUMP 268
*LABEL 271
*JNZ,-1(N,274)
INAM9='VIBR
INAM7='VBVL
INAM8='VBMD
*DCALL(MDDL METH)
*JUMP 264
*LABEL 274
*JNZ,-1(N,275)
INAM9='VIBR
INAM7='VBVL
INAM8='VBMD
*DCALL(MDDL METH)
*JUMP 268
*LABEL 275
*XQT EXIT
*LABEL 268
*RETURN
*
*
$----- 8
$
*(CALC KM)
ENDCKM$PERFORM STATIC AND/OR DYNAMIC ANALYSES OF INITIAL STRUCTURE
*XQT E
*XQT EKS
*XQT K
*JNZ(DISP,156)
*JNZ(STRD,156)
*JUMP 157

```

APPENDIX

```

*LABEL 156
*XQT RSI
!A=NLST
*LABEL 5666
*JLZ,-1(A,5667)
!NNST=NLST-A
*XQT SSOL$CALCULATE DISPLACEMENTS
  RESET SET="NNST"
*JUMP 5666
*LABEL 5667
!A=FREE()
!NNST=FREE()
*LABEL 157
*XQT U1
!NN3=TOC,N3(1,K,SPAR,MASK,MASK)
!NN4=TOC,N4(1,K,SPAR,MASK,MASK)
*JZ(STRD,150)
!A=NLST
*LABEL 7666
*JLZ,-1(A,7667)
!NNST=NLST-A
!NAM1='STAT
!NAM2='DISP
!NAM3="NNST"
!NAM4=1
!OLI8=1
!NI1=-1000 $SET NI=-1000 FOR UNPERTURBED STRESS CALCULATIONS
*DCALL(CALC STRS)$CALCULATE STRESSES
*JUMP 7666
*LABEL 7667
!A=FREE()
!NNST=FREE()
!NAM1=FREE()
!NAM2=FREE()
!NAM3=FREE()
!NAM4=FREE()
!OLI8=OTI8
*LABEL 150
*JNZ(DISP,193)
*JNZ(STRD,193)
*XQT RSI
*XQT SSOL
*LABEL 193
*JZ(BCUD,151)
*XQT GSF
  RESET EMBED=1
*XQT KG
*XQT EIG$PERFORM BUCKLING ANALYSIS
  RESET CONV=1.-10
  RESET PROB=BUCK
  RESET INIT="MMDB",NREQ="MRQB"
  RESET NDYN=20
*XQT EIG
  RESET CONV=1.-10,PROB=BUCK,NREQ="MRQB",NDYN=20
*LABEL 151
*JZ(VIBD,152)
*XQT E4$PERFORM VIBRATION ANALYSIS
  RESET NMODES="MMOD",NREQ="MREQ"
*LABEL 152
*XQT DCU
*JNZ(DISP,159)
*JNZ(STRD,159)
*JUMP 158
*LABEL 159
!A=NLST
*LABEL 6666
*JLZ,-1(A,6667)
!NNST=NLST-A
$RENAME ALL DATA SETS AS INDICATED BELOW
CHANGE 1 STAT DISP "NNST" 1,PREV DISP "NNST" 1
*JUMP 6666
*LABEL 6667
*LABEL 159
!A=FREE()
!NNST=FREE()
*LABEL 158
CHANGE 1 DEM DIAG 0 0,DEMP DIAG 0 0
CHANGE 1 K SPAR "NN3" "NN4",KP SPAR "NN3" "NN4"

```

APPENDIX

```

CHANGE 1 INV K 1 0,INV KP 1 0
CHANGE 1 XINV K 1 0,XINV KP 1 0
*JZ(BUGD,154)
!NNN3=TOC,N3(1,KG,SPAR,MASK,MASK)
!NNN4=TOC,N4(1,KG,SPAR,MASK,MASK)
CHANGE 1 KG SPAR "NNN3" "NNN4",KGP SPAR "NNN3" "NNN4"
CHANGE 1 BUCK EVAL 1 1,PREV BKVL 1 1
CHANGE 1 BUCK MODE 1 1,PREV BKMD 1 1
*LABEL 154
*JZ(VIBD,155)
CHANGE 1 VIBR EVAL 1 1,PREV VBVL 1 1
CHANGE 1 VIBR MODE 1 1,PREV VBMD 1 1
*LABEL 155
*RETURN
*
*
$
$----- 9
$
*{DKDV DMDV)          ENDDKM$CALCULATE DK/DV,DM/DV, AND DKG/DV
!IIV=0
!IEV=NDV
*LABEL 803
*JLZ,-1(!IEV,804)$LOOP OVER DESIGN VARIABLES
!IIV=IIV+1
!DVP=DS,1,"IIV",1(1,DESV,CNMN,MASK,MASK)
!DV=DR*DVP+DVP
!AQ=1./DR
!AQ=AQ/DVP
!BQ=-1.*AQ
*XQT AUS
TABLE,U(NJ="NDV");DESV CNMN$ INCREMENT DESIGN VARIABLE
OPER=XSUM
J="IIV";"DV"
*DCALL(TRNS DESV)
*DCALL(LINK POLY)
*XQT DCU
PRINT 1 LNK DV
*XQT TAB
UPDATE=1
*JZ(NDVA,800)
!IEV=NDVA
*DCALL(DVA FDD)$ UPDATES ALL MEMBRANE PROPS. TO CURRENT VALUE
*LABEL 800
*JZ(NPDV,801)
!IEV=NPDV
*DCALL(DVP FDD)$ UPDATES ALL PLATE PROPS. TO CURRENT VALUES
*LABEL 801
*JZ(NBDV,802)
!IEV=NBDV
*DCALL(DVB FDD)$ UPDATES ALL BEAM PROPS. TO CURRENT VALUES
*LABEL 802
!NN3=TOC,N3(1,KP,SPAR,MASK,MASK)
*XQT E
*XQT EKS
*XQT K
  RESET OUTLIB= "OLIB"
*XQT AUS
DEFINE KN= "OLIB" K SPAR
DEFINE KD=1 KP SPAR
DK=SUM("AQ" KN,"BQ" KQ)
DEFINE A=DEMP DIAG 0 0
DEFINE B=DEM DIAG 0 0
DMDV DIAG 0 "IIV"=SUM("AQ" B,"BQ" A)
*XQT DCU
$PRINT 15 K SPAR
$PRINT 1 KP SPAR
CHANGE 1 DK MASK MASK MASK, DKDV SPAR "NN3" "IIV"
$PRINT 1 DKDV SPAR "NN3" "IIV"
*JZ(BUGD,294)
*XQT RSI
RESET NUK="OLIB"
*XQT SSOL
RESET KLIB="JLIB"
RESET KLIB=1
*XQT GSF
RESET EMBED=1
*LABEL 292
*JNZ(BUGD,293)

```


APPENDIX

```

*JNZ(V18D,293)
*JUMP 294
*LABEL 293
*XQT KG
*XQT AUS
DEFINE KGN=KG SPAR "NNN3" "NNN4"
DEFINE KGD=KGP SPAR "NNN3" "NNN4"
DKG=SUM("AQ" KGN,"BQ" KGD)
*XQT DCU
CHANGE 1 DKG SPAR "NNN3" "NNN4",DKG SPAR "NNN3" "IIV"
*LABEL 294
*XQT AUS
TABLE,U(INJ="NDV");DESV CNMNS SET ORIGINAL VALUE OF DESIGN VARIABLE
OPER=XSUM
J="IIV";"JVP"
INIEV=IEV
*JNZ(NIEV,805)
*DCALL(TRNS DESV)
*DCALL(LINK POLY)
*LABEL 805
*DCALL(DCU EFIL)
*JUMP 803
*LABEL 804
!DI1=FREE()
!DI2=FREE()
!DI3=FREE()
!DI4=FREE()
!DI5=FREE()
!DI6=FREE()
!DI7=FREE()
!DI8=FREE()
!DI9=FREE()
!DI10=FREE()
!DI11=FREE()
!DI113=FREE()
!DI14=FREE()
!DI15=FREE()
!DI16=FREE()
!DI17=FREE()
!DI18=FREE()
!DI19=FREE()
!DI20=FREE()
!DI21=FREE()
!DI22=FREE()
!DI23=FREE()
!DI24=FREE()
!DI25=FREE()
*RETURN
*
*
$
$----- 10
$
*(CALC DUDV)
$ENDDUV$CALCULATE DU/DAPV
!A=NLST
*LABEL 5666
*JLZ,-1(A,5667) $LOOP OVER LOAD SETS
!NNST=NLST-A
!IIV=0
!IEV=NDV
*LABEL 7777
*JLZ,-1(IEV,7778)$LOOP OVER DESIGN VARIABLES
!IIV=IIV+1
!NNST
*XQT AUS
DEFINE U=PREV DISP "NNST" 1
DEFINE DK1=DKDV SPAR "NN3" "IIV"
DKU=PROD(DK1,U)
*XQT DCU
$PRINT 1 DKU MASK
$PRINT 1 PREV DISP "NNST" 1
*XQT AUS
!IS=IIV-1
*JGZ(IS,825)
APPL FORCE 50=UNION(-1. DKU)
*XQT DCU
*JUMP 826
*LABEL 825
DEFINE AF2=APPL FORCE 50

```

APPENDIX

30

APPENDIX

```

$
$----- 12
$
*(DVA FDD)          ENDFDAS UPDATE ALL MEMBRANE DESIGN VARIABLES
INIIV=IEV
*LABEL 900
*JLZ,-1(IEV,901)
INNIV=NIIV-IEV
ITYP=DS,1,"NNIV",1(1,DVA,DFN,MASK,MASK)
ISECT=DS,4,"NNIV",1(1,DVA,DFN,MASK,MASK)
!DDVP=DS,1,"NNIV",1(1,LNK,DV,MASK,MASK)
IRR=DDVP
*DCALL(FD LP1)
*JUMP 900
*LABEL 901
ISECT=FREE()
ITYP=FREE()
IRR=FREE()
!DDVP=FREE()
INNIV=FREE()
*RETURN
*          ENDFDA
$
$-----
$
*(FD LP1)          ENDFDL
$          UPDATE MEMBRANE AND ROD ELEMENT PROPERTIES TO
$          VALUE RR WHICH IS DETERMINED PREVIOUSLY
INTYP="TYP"
*JNZ,-1(INTYP,1802)$TEST IF E23 ELEMENT
$SET ROD SECTION PROPERTIES
E23 SECTION PROPERTIES
"SECT" "RR"
*JUMP 1803
*LABEL 1802
INTYP="TYP"
*JNZ,-2(INTYP,1804)$TEST IF E31 ELEMENT
$SET MEMBRANE SECTION PROPERTIES
SA
"SECT" "RR"
*JUMP 1803
*LABEL 1804
INTYP="TYP"
*JNZ,-5(INTYP,1807)$TEST IF E41 ELEMENT
$SET MEMBRANE SECTION PROPERTIES
SA
"SECT" "RR"
*JUMP 1803
*LABEL 1807
INTYP="TYP"
*JNZ,-8(INTYP,1810)$TEST IF E44 ELEMENT
$SET SHEAR PANEL PROPERTIES
SB
"SECT" "RR"
*JUMP 1803
*LABEL 1810
*XQT EXIT
*LABEL 1803
*RETURN
*          ENDFDL
$
$----- 13
$
*(DVP FDD)          ENDFDPS UPDATE ALL PLATE DESIGN VARIABLES
!NIIV=IEV
*LABEL 902
*JLZ,-1(IEV,903)
INNIV=NIIV-IEV
ITYP=DS,1,"NNIV",1(1,DVP,DFN,MASK,MASK)
ISECT=DS,4,"NNIV",1(1,DVP,DFN,MASK,MASK)
!NRIV=NNIV+NDVA
!DDVP=DS,1,"NRIV",1(1,LNK,DV,MASK,MASK)
IRR=DDVP
*DCALL(FD LP2)
*JUMP 902
*LABEL 903
!NIIV=FREE()
!ITYP=FREE()

```

APPENDIX

```

ISECT=FREE()
INRIV=FREE()
IDDVP=FREE()
IKR=FREE()
*RETURN
*
*                                ENDFDP
$
$-----
$
*(FD LP2)                                ENOFD2
INTYP="TYP"
*JZ,-3(NTYP,600)$TEST IF E32 ELEMENT
*JZ,-1(NTYP,600)$TEST IF E33 ELEMENT
*JZ,-2(NTYP,600)$TEST IF E42 ELEMENT
*JZ,-1(NTYP,600)$TEST IF E43 ELEMENT
*XQT EXIT
*LABEL 600
*DCALL(SA TYP)
*RETURN
*
*                                ENFD2
$
$-----
$
*(SA TYP)                                ENDSAT
SA
"SECT" "RR"
*RETURN
*
*                                ENDSAT
$
$-----
$
*(DVB FDD)                                ENDFDB$ UPDATE ALL BEAM DESIGN VARIABLES
ISEC1=0
INIIV=IEV
*LABEL 904
*JLZ,-1(IEV,905)
INNIV=INIIV-IEV
ISECT=DS,4,"NNIV",1(1,DVB,DFN,MASK,MASK)
INKIV=INNIV+NDVA+NPOV
IDDVP=DS,1,"NRIV",1(1,LNK,DV,MASK,MASK)
IKR=DDVP
*UCALL(BA TYP2)
*JUMP 904
*LABEL 905
ISEC1=FREE()
INIIV=FREE()
INNIV=FREE()
ISECT=FREE()
INRIV=FREE()
IDDVP=FREE()
IKR=FREE()
*RETURN
*
*                                ENDFDB
$
$-----
$
*(BA TYP2)                                ENDBA2$ UPDATES BEAM DESIGN VARIABLES
ISEC2=SECT-SEC1
*JZ(SEC2,1000)
IDI1=DS,26,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI2=DS,27,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI3=DS,28,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI4=DS,29,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI5=DS,30,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI6=DS,31,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI7=DS,4,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI8=DS,5,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI9=DS,6,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI10=DS,7,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI11=DS,8,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI12=DS,9,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI13=DS,10,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI14=DS,11,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI15=DS,12,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI16=DS,13,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI17=DS,14,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI18=DS,15,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI19=DS,16,1,"SECT"(1,BA,BTAB,MASK,MASK)

```

14

APPENDIX

```

IDI20=DS,18,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI21=DS,19,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI22=DS,20,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI23=DS,21,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI24=DS,22,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI25=DS,23,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI26=DS,24,1,"SECT"(1,BA,BTAB,MASK,MASK)
IDI27=DS,25,1,"SECT"(1,BA,BTAB,MASK,MASK)
*LABEL 1000
*XQT U1
*$SHOW
*XQT TAB
UPDATE=1
!DVB1=DS,1,"NNIV",1(1,DVB,DFN,MASK,MASK)
!DVB2=DS,5,"NNIV",1(1,DVB,DFN,MASK,MASK)
*JNZ,-1(DVB2,1001)
!DI1=RR
*JUMP 1007
*LABEL 1001
*JNZ,-1(DVB2,1002)
!DI2=RR
*JUMP 1007
*LABEL 1002
*JNZ,-1(DVB2,1003)
!DI3=RR
*JUMP 1007
*LABEL 1003
*JNZ,-1(DVB2,1004)
!DI4=RR
*JUMP 1007
*LABEL 1004
*JNZ,-1(DVB2,1005)
!DI5=RR
!DI6=ID6
*JUMP 1007
*LABEL 1005
*JNZ,-1(DVB2,1006)
!DI6=RR
*JUMP 1007
*LABEL 1006
*JNZ,-1(DVB2,1030)
!DI7=RR
*JUMP 1007
*LABEL 1030
*JNZ,-1(DVB2,1031)
!DI8=RR
*JUMP 1007
*LABEL 1031
*JNZ,-1(DVB2,1032)
!DI9=RR
*JUMP 1007
*LABEL 1032
*JNZ,-1(DVB2,1033)
!DI10=RR
*JUMP 1007
*LABEL 1033
*JNZ,-1(DVB2,1034)
!DI11=RR
*JUMP 1007
*LABEL 1034
*JNZ,-1(DVB2,1035)
!DI12=RR
*JUMP 1007
*LABEL 1035
*JNZ,-1(DVB2,1036)
!DI13=RR
*JUMP 1007
*LABEL 1036
*JNZ,-1(DVB2,1037)
!DI14=RR
*JUMP 1007
*LABEL 1037
*JNZ,-1(DVB2,1038)
!DI15=RR
*JUMP 1007
*LABEL 1038

```

APPENDIX

```

*JNZ,-1(DVB2,1039)
!DI16=RR
*JUMP 1007
*LABEL 1039
*JNZ,-1(DVB2,1040)
!DI17=RR
*JUMP 1007
*LABEL 1040
*JNZ,-1(DVB2,1041)
!DI18=RR
*JUMP 1007
*LABEL 1041
*JNZ,-1(DVB2,1042)
!DI19=RR
*JUMP 1007
*LABEL 1042
*JNZ,-1(DVB2,1043)
!DI20=RR
*JUMP 1007
*LABEL 1043
*JNZ,-1(DVB2,1044)
!DI21=RR
*JUMP 1007
*LABEL 1044
*JNZ,-1(DVB2,1045)
!DI22=RR
*JUMP 1007
*LABEL 1045
*JNZ,-1(DVB2,1046)
!DI23=RR
*JUMP 1007
*LABEL 1046
*JNZ,-1(DVB2,1047)
!DI24=RR
*JUMP 1007
*LABEL 1047
*JNZ,-1(DVB2,1048)
!DI25=RR
*JUMP 1007
*LABEL 1048
*JNZ,-1(DVB2,1049)
!DI26=RR
*JUMP 1007
*LABEL 1049
*JNZ,-1(DVB2,1050)
!DI27=RR
*JUMP 1007
*LABEL 1050
*XQT EXIT
*LABEL 1007
*JNZ,-1(DVB1,1008)
*DCALL(TUBE BEAM)
*JUMP 1016
*LABEL 1008
*JNZ,-1(DVB1,1009)
*DCALL(BOX BEAM)
*JUMP 1016
*LABEL 1009
*JNZ,-1(DVB1,1010)
*DCALL(TEE BEAM)
*JUMP 1016
*LABEL 1010
*JNZ,-1(DVB1,1011)
*DCALL(ANG BEAM)
*JUMP 1016
*LABEL 1011
*JNZ,-1(DVB1,1012)
*DCALL(WFL BEAM)
*JUMP 1016
*LABEL 1012
*JNZ,-1(DVB1,1013)
*DCALL(CHN BEAM)
*JUMP 1016
*LABEL 1013
*JNZ,-1(DVB1,1014)
*DCALL(ZEE BEAM)
*JUMP 1016
*LABEL 1014

```

APPENDIX

```

*JNZ,-1(DVB1,1015)
*DCALL(GIVN BEAM)
*JUMP 1016
*LABEL 1015
*JNZ,-1(DVB1,1020)
*DCALL(DSY BEAM)
*JUMP 1016
*LABEL 1020
*XQT EXIT
*LABEL 1016
ISEC1="SECT"
ISEC2=FREE()
IDVB1=FREE()
IDVB2=FREE()
*RETURN
*
*                                     ENDBA2
$
$-----
$
* (TUBE BEAM)                       ENDTUB$ SET TUBE SECTION PROPERTIES
BA
TUBE "SECT","DI1","DI2"
*RETURN
*
*                                     ENDTUB
$
$-----
$
* (BOX BEAM)                       ENDBJB$ SET BOX SECTION PROPERTIES
BA
BOX "SECT","DI1","DI2","DI3","DI4"
*RETURN
*
*                                     ENDBJB
$
$-----
$
* (TEE BEAM)                       ENDTB$ SET TEE SECTION PROPERTIES
BA
TEE "SECT","DI1","DI2","DI3","DI4"
*RETURN
*
*                                     ENDTB
$
$-----
$
* (ANG BEAM)                       ENDANB$ SET ANG SECTION PROPERTIES
BA
ANG "SECT","DI1","DI2","DI3","DI4"
*RETURN
*
*                                     ENDANB
$
$-----
$
* (WFL BEAM)                       ENDWFB$ SET WFL SECTION PROPERTIES
BA
WFL "SECT","DI1","DI2","DI3","DI4","DI5","DI6"
*RETURN
*
*                                     ENDWFB
$
$-----
$
* (CHN BEAM)                       ENDCHB$ SET CHN SECTION PROPERTIES
BA
CHN "SECT","DI1","DI2","DI3","DI4","DI5","DI6"
*RETURN
*
*                                     ENDCHB
$
$-----
$
* (ZEE BEAM)                       ENDZEB$ SET ZEE SECTION PROPERTIES
BA
ZEE "SECT","DI1","DI2","DI3","DI4","DI5","DI6"
*RETURN
*
*                                     ENDZEB
$
$-----
$
* (GIVN BEAM)                      ENDGVB$ SET GIVN SECTION PROPERTIES
BA

```

APPENDIX

```

GIVN "SECT","DI7","DI8","DI9","DI10","DI11","DI12","DI13","DI14"%
"DI15","DI16"
*RETURN
*
*                               ENOGVB
$
$-----
$
$(DSY BEAM)                     ENDDSY
BA
DSY "SECT","DI7","DI8","DI9","DI10","DI11","DI12","DI13"%
"DI17","DI18","DI19","DI20","DI21","DI22","DI23","DI24"%
"DI25","DI26","DI27"
*RETURN
*                               ENDDSY
$
$----- 15
$
$(EFIL DCU)                     ENDEDU
*XQT DCU
*JZ(DE21,8010)
DISABLE 1 E21 EFIL MASK MASK
CHANGE 1 EE21 EEIL 1 "IIV",E21 EFIL 1 2
*LABEL 8010
*JZ(DE23,8011)
DISABLE 1 E23 EFIL MASK MASK
CHANGE 1 EE23 EEIL 3 "IIV",E23 EFIL 3 2
*LABEL 8011
*JZ(DE31,8012)
DISABLE 1 E31 EFIL MASK MASK
CHANGE 1 EE31 EEIL 6 "IIV",E31 EFIL 6 3
*LABEL 8012
*JZ(DE32,8013)
DISABLE 1 E32 EFIL MASK MASK
CHANGE 1 EE32 EEIL 7 "IIV",E32 EFIL 7 3
*LABEL 8013
*JZ(DE33,8014)
DISABLE 1 E33 EFIL MASK MASK
CHANGE 1 EE33 EEIL 8 "IIV",E33 EFIL 8 3
*LABEL 8014
*JZ(DE41,8015)
DISABLE 1 E41 EFIL MASK MASK
CHANGE 1 EE41 EEIL 9 "IIV",E41 EFIL 9 4
*LABEL 8015
*JZ(DE42,8016)
DISABLE 1 E42 EFIL MASK MASK
CHANGE 1 EE42 EEIL 10 "IIV",E42 EFIL 10 4
*LABEL 8016
*JZ(DE43,8017)
DISABLE 1 E43 EFIL MASK MASK
CHANGE 1 EE43 EEIL 11 "IIV",E43 EFIL 11 4
*LABEL 8017
*JZ(DE44,8018)
DISABLE 1 E44 EFIL MASK MASK
CHANGE 1 EE44 EEIL 12 "IIV",E44 EFIL 12 4
*LABEL 8018
*RETURN
*                               ENDEDU
$
$-----
$
$(DCU EFIL)                     ENDDDEL
*XQT DCU
*JZ(DE21,8000)
CHANGE 1 E21 EFIL 1 2,EE21 EEIL 1 "IIV"
*LABEL 8000
*JZ(DE23,8001)
CHANGE 1 E23 EFIL 3 2,EE23 EEIL 3 "IIV"
*LABEL 8001
*JZ(DE31,8002)
CHANGE 1 E31 EFIL 6 3,EE31 EEIL 6 "IIV"
*LABEL 8002
*JZ(DE32,8003)
CHANGE 1 E32 EFIL 7 3,EE32 EEIL 7 "IIV"
*LABEL 8003
*JZ(DE33,8004)
CHANGE 1 E33 EFIL 8 3,EE33 EEIL 8 "IIV"
*LABEL 8004
*JZ(DE41,8005)

```


APPENDIX

```

CHANGE 1 E41 EFIL 9 4,EE41 EEIL 9 "IIV"
*LABEL 8005
*JZ(DE42,8006)
CHANGE 1 E42 EFIL 10 4,EE42 EEIL 10 "IIV"
*LABEL 8006
*JZ(DE43,8007)
CHANGE 1 E43 EFIL 11 4,EE43 EEIL 11 "IIV"
*LABEL 8007
*JZ(DE44,8008)
CHANGE 1 E44 EFIL 12 4,EE44 EEIL 12 "IIV"
*LABEL 8008
*RETURN
*
*                               ENDDCL
$
$----- 16
$
*(STRS DERV)                               ENDSTD
$$
$$ CALCULATE STRESS DERIVATIVES
!IE1=NLST$LOOP OVER LOAD SETS
*LABEL 2102
*JLZ,-1(!IE1,2103)
!NIE1=NLST-IE1$LOAD SET NUMBER
!IEV=NDV
!IS=0
!ISS=0
*LABEL 2100
*JLZ,-1(!IEV,2101)$LOOP OVER DESIGN VARIABLES
!IEV=NDV-IEV
!DVP=DS,1,"IIV",1(1,DESV,CNMN,MASK,MASK)
*DCALL(EFIL DCU)
!DELV=DR*DVP
!AQ=1./DELV
!BQ=-1.*AQ
!NCA=LCAS$ LOOP OVER LOAD CASES
*LABEL 3000
*JLZ,-1(!NCA,3001)
*XQT AUS
!NCAS=LCAS-NCA
!N11=IIV*LCAS-LCAS+NCAS
DEFINE U1=PREV DISP "NIE1" 1 "NCAS","NCAS"
DEFINE U2=DUDV CAPV "NIE1" 1 "N11","N11"
*JGZ(!IS,3825)
U DELU "NIE1" 1 =SUM(U1,"DELV" U2)
!IS=IS+1
*JUMP 3826
*LABEL 3825
DEFINE TEMP=U DELU "NIE1" 1
DU=SUM(U1,"DELV" U2)
UDEL TMP=UNION(TEMP,DU)
*XQT DCU
CHANGE 1 UDEL TMP 1 1,U DELU "NIE1" 1
*LABEL 3826
*JUMP 3000
*LABEL 3001
!NAM1='U
!NAM2='DELU
!NAM3="NIE1"
!NAM4='MASK
*DCALL(CALC STRS)
!IS=0
!ISS=ISS+1
*DCALL(DCU EFIL)
*JUMP 2100
*LABEL 2101
!IEV=FREE()
!IEV=FREE()
!IEV=FREE()
!NCAS=FREE()
!N11=FREE()
!IEV=FREE()
!NCA=FREE()
!NAM1=FREE()
!NAM2=FREE()
!NAM3=FREE()
!NAM4=FREE()
*XQT AUS
OUTLIB=1
*JZ(DE21,2104)

```

APPENDIX

```

!IE=NE21
!NAM5='E21
*DCALL(DSIG EIJ)
*LABEL 2104
*JZ(DE23,2105)
!IE=DE23
!NAM5='E23
*DCALL(DSIG EIJ)
*LABEL 2105
*JZ(DE31,2106)
!IE=DE31
!NAM5='E31
*DCALL(DSIG EIJ)
*LABEL 2106
*JZ(DE32,2107)
!IE=DE32
!NAM5='E32
*DCALL(DSIG EIJ)
*LABEL 2107
*JZ(DE33,2108)
!IE=DE33
!NAM5='E33
*DCALL(DSIG EIJ)
*LABEL 2108
*JZ(DE41,2109)
!IE=DE41
!NAM5='E41
*DCALL(DSIG EIJ)
*LABEL 2109
*JZ(DE42,2110)
!IE=DE42
!NAM5='E42
*DCALL(DSIG EIJ)
*LABEL 2110
*JZ(DE43,2111)
!IE=DE43
!NAM5='E43
*DCALL(DSIG EIJ)
*LABEL 2111
*JZ(DE44,2112)
!IE=DE44
!NAM5='E44
*DCALL(DSIG EIJ)
*LABEL 2112
*JUMP 2102
*LABEL 2103
!NIE1=FREE()
!IE1=FREE()
!IE=FREE()
!NAM5=FREE()
*RETURN
*
                                ENDSTD
$
$-----
$
$(CALC STRS)                ENDCCS
$$                          $$ CALCULATES STRESSES OF DESIGN VARIABLES IN
$$                          $$ ES FORMAT TO BE USED IN STRESS DERIVATIVE
$$                          $$ AND CONSTRAINT CALCULATIONS.
*XQT ES
  OUTLIB= "OLIB"
  U= "NAM1" "NAM2" "NAM3" "NAM4"
PMODE= 2
NODES=0
*JZ(NE21,2500)
!IE=NE21
!NAM5='E21
*DCALL(STRS GRP)
*LABEL 2500
*JZ(DE23,2501)
!IE=DE23
!NAM5='E23
*DCALL(STRS GRP)
*LABEL 2501
*JZ(DE31,2502)
!IE=DE31
!NAM5='E31

```

APPENDIX

```

*DCALL(STRS GRP)
*LABEL 2502
*JZ(DE32,2503)
!IE=DE32
!NAM5='E32
*DCALL(STRS GRP)
*LABEL 2503
*JZ(DE33,2504)
!IE=DE33
!NAM5='E33
*DCALL(STRS GRP)
*LABEL 2504
*JZ(DE41,2505)
!IE=DE41
!NAM5='E41
*DCALL(STRS GRP)
*LABEL 2505
*JZ(DE42,2506)
!IE=DE42
!NAM5='E42
*DCALL(STRS GRP)
*LABEL 2506
*JZ(DE43,2507)
!IE=DE43
!NAM5='E43
*DCALL(STRS GRP)
*LABEL 2507
*JZ(DE44,2508)
!IE=DE44
!NAM5='E44
*DCALL(STRS GRP)
*LABEL 2508
*DCALL(CAL STRS)
!IE=FREE()
!NAM5=FREE()
*RETURN
*
$
$-----
$
*(CAL STRS)
*JZ(NE21,2500)
!IE=NE21
!NAM5='E21
*DCALL(STRS CALC)
*LABEL 2500
*JZ(DE23,2501)
!IE=DE23
!NAM5='E23
*DCALL(STRS CALC)
*LABEL 2501
*JZ(DE31,2502)
!IE=DE31
!NAM5='E31
*DCALL(STRS CALC)
*LABEL 2502
*JZ(DE32,2503)
!IE=DE32
!NAM5='E32
*DCALL(STRS CALC)
*LABEL 2503
*JZ(DE33,2504)
!IE=DE33
!NAM5='E33
*DCALL(STRS CALC)
*LABEL 2504
*JZ(DE41,2505)
!IE=DE41
!NAM5='E41
*DCALL(STRS CALC)
*LABEL 2505
*JZ(DE42,2506)
!IE=DE42
!NAM5='E42
*DCALL(STRS CALC)
*LABEL 2506
*JZ(DE43,2507)
!IE=DE43

```

ENDCCS

ENDCALL

APPENDIX

```

INAM5='E43
*DCALL(STRS CALC)
*LABEL 2507
*JZ(DE44,2508)
!IE=DE44
INAM5='E44
*DCALL(STRS CALC)
*LABEL 2508
!IE=FREE()
INAM5=FREE()
*RETURN
*                                ENDCALL
$
$-----
$
*(STRS CALC)                ENDSTS
!NAMJ=IE
*LABEL 2500
*JLZ,-1(NAMJ,2501)
!IV=IE-NAMJ
*DCALL(STRS BLD)
*JUMP 2500
*LABEL 2501
!NAMJ=IE
*LABEL 2600
*JLZ,-1(NAMJ,2601)
!IV=IE-NAMJ
*DCALL(CHNG NAME)
*JUMP 2600
*LABEL 2601
!NAMJ=FREE()
!IV=FREE()
*RETURN
*                                ENDSTS
$
$-----
$
*(STRS BLD)                STRSB
*JLZ(N11,5000)
*XQT AUS
INLIB="OLIB"
OUTLIB="OLIB"
$!IS
$!N11
$!OLIB
$!NAM5
$!NIE1
$!ISS
$!IV
DEFINE ENEW="OLIB" ES "NAM5" "NIE1" "IV"
$ENEW=UNION(ETMP)
*JZ(ISS,2202)
DEFINE EOLD="OLIB" EST "NAM5" "NIE1" "IV"
$EOLD=UNION(ESS)
EST1 "NAM5" "NIE1" "IV" = UNION(EOLD,ENEW)
*XQT DCU
CHANGE "OLIB" EST1 "NAM5" "NIE1" "IV",EST "NAM5" "NIE1" "IV"
$ TUC "OLIB"
$PRINT "OLIB" EST "NAM5" "NIE1" "IV"
*JUMP 3001
*LABEL 2202
EST "NAM5" "NIE1" "IV" = UNION(ENEW)
*XQT DCU
$ TUC "OLIB"
$PRINT "OLIB" EST "NAM5" "NIE1" "IV"
*LABEL 3001
*LABEL 5000
*RETURN
*                                STRSB
$
$-----
$
*(STRS GRP)                ENDGRP
!NAMJ=IE
*LABEL 1000
*JLZ,-1(NAMJ,1001)
!IV=IE-NAMJ
"NAM5" "IV"

```

APPENDIX

```

*JUMP 1000
*LABEL 1001
*RETURN
*
*                               ENDGRP
$
$-----
$
$(CHNG NAME)                ECHG
!N11
*JLZ(N11,5000)
!IIIV
!IEEV
*JZ(IEEV,3000)
*JUMP 5000
*LABEL 3000
*XQT DCU
CHANGE "DLIB" EST "NAM5" "NIE1" "IV",ES "NAM5" "NIE1" "IV"
*LABEL 5000
*RETURN
*
*                               ECHG
$
$-----
$
$(DSIG EIJ)                ENDDSG
$$                          $$ SUBPROGRAM USED TO CALCULATE STRESS DERIVATIVES
$$                          $$ USING PRASAD'S METHOD
!IE2=IE
*LABEL 2200
*JLZ,-1(IE,2201)$LOOP OVER GROUPS
!NN1=1
!IV=IE2-IE$GRJUP NUMBER
!IEEV=NDV
*LABEL 2100
*JLZ,-1(IEEV,2101)$LOOP OVER DESIGN VARIABLES
!IIIV=NDV-IEEV
!DVP=DS,1,"IIIV",1(1,DESV,CNMN,MASK,MASK)
!DELV=DR*DVP
!AQ=1./DELV
!BQ=-1.*AQ
INCA=LCAS$ LOOP OVER LOAD CASES
*LABEL 3000
*XQT AUS
*JLZ,-1(NCA,3001)
!NCAS=LCAS-NCA
!N11=IIIV*LCAS-LCAS+NCAS
*XQT AUS
DEFINE STRP=1 ES "NAM5" "NIE1" "IV" "NCAS","NCAS"
DEFINE STRN="DLIB" ES "NAM5" "NIE1" "IV" "N11","N11"
BB1=SUM("AQ" STRN,"BQ" STRP)
$*XQT DCU
$ TOC 1
$$PRINT 1 BB1 MASK MASK MASK
$*XQT AUS
*JZ,-1(NN1,2202)
!NN1=NN1+1
DEFINE DSG=DSIG "NAM5" "NIE1" "IV"
$DSG=UNION(DSG)
$DG=UNION(DSG,BB1)
DTMP "NAM5" "NIE1" "IV"=UNION(DSG,BB1)
*XQT DCU
CHANGE 1 DTMP "NAM5" "NIE1" "IV",DSIG "NAM5" "NIE1" "IV"
$ TOC 1
$PRINT 1 DSIG "NAM5" "NIE1" "IV"
*XQT AUS
*JUMP 3000
*LABEL 2202
!NN1=NN1+3
DSIG "NAM5" "NIE1" "IV"=UNION(BB1)
$*XQT DCU
$ TOC 1
$ $PRINT 1 DSIG "NAM5" "NIE1" "IV"
$*XQT AUS
*JUMP 3000
*LABEL 3001
*JUMP 2100
*LABEL 2101
*JUMP 2200
*LABEL 2201

```

APPENDIX

```

!IE2=FREE()
!IV=FREE()
!IIV=FREE()
!IEV=FREE()
!NCAS=FREE()
!N11=FREE()
!NCA=FREE()
*RETURN
*          ENDDSG
$
$----- 17
$
*(NELS METH)          ENDNLM
$$          $$ CALCULATES DERIVATIVES OF VIB. OR BUCK. MODES
$$          $$ AND/OR LOADS ANALYTICALLY.
*DCALL(BOND COND)
!JDF=DS,2,1,1(1,JDF1,BTAB,1,8)
!NNMD=MODE
!TT1=EQUAL(NAM9,MODE1)
!TT2=EQJAL(NAM9,MODE2)
*XQT AUS
!NNDV=NDDV
*LABEL 4109
*JLZ,-1(NNDV,4108)
!IDV=NDDV-NNDV
TABLE(NJ=1,NI="MODE");DVAL "NAM9" 1 "IDV"
!J=1;J=1;0.
*JUMP 4109
*LABEL 4108
*LABEL 2010
*JLZ,-1(NNMD,2000)$LOOP ON MODES
!NM=MODE-NNMD
!NMI=NM
!MODN=DS,1,"NM",1(1,MODE,NUM,MASK,MASK)
!ISKP=MODN-MODN
!IBSE=MODN-1
$          FIND EIGENVALUE
*XQT AUS
DEFINE L=PREV "NAM7" 1 1
TABLE(NI=1,NJ=1);LAM1 AUS 1 1
TRAN(SOURCE=L,SBASE="IBSE",ILIM=1,SSKIP="ISKP")
*XQT U1
!LAMM=DS,1,1,1(1,LAM1,AUS,1,1)
!LAMN=.5*LAMM
*XQT AUS
DEFINE PP=PREV "NAM8" 1 1 "MODN","MODN"
PHI=UNION(1.0 PP)
*XQT DCU
PRINT 1 PHI
*XQT AUS
*JZ(TT1,4100)
LDEM DIAG 0 0=UNION("LAMM" DEMP)
DEFINE LDEM=LDEM DIAG
ASP SPAR=SUM(KP,-1.0 LDEM)
*LABEL 4100
*JZ(TT2,4101)
!LAMN=.5*LAMM
LDEM DIAG 0 0=SUM("LAMN" KGP,"LAMN" KGP)
ASP SPAR=SUM(KP,1.0 LDEM)
*LABEL 4101
PONE=UNION("LAMM" PHI)
!NNDV=NDDV
*LABEL 2006
*JLZ,-1(NNDV,2001)$LOOP OVER DESIGN VARIABLES
!IDV=NDDV-NNDV
!JDF=DS,2,1,1(1,JDF1,BTAB,1,8)
!NDF=JDF*JDF
*JZ(TT1,2003)
DEFINE DM=DMDV DIAG 0 "IDV"
*LABEL 2003
*JZ(TT2,2004)
DEFINE DM=DKG SPAR "NDF" "IDV"
*LABEL 2004
DEFINE DK=DKDV SPAR "NDF" "IDV"
PTWO=PROD(DM,PONE)
PTRI=PROD(DK,PHI)

```

APPENDIX

```

PFOR=XTYD(PHI,PTRI)
PFIV=XTYD(PHI,PTWO)
*JZ(TT1,4500)
PSIX=SUM(PFOR,-1. PFIV)
*LABEL 4500
*JZ(TT2,4501)
PSIX=SUM(1. PFOR,1. PFIV)
*LABEL 4501
EDER DIAG "MODN" "IDV"=UNION(1.0 PSIX)
IEDDG=DS,1,1,1(1,EDER,DIAG,"MODN","IDV")
TABLE,U(NJ=1,N1="MODE");DVAL "NAM9" 1 "IDV"
OPERATION=XSUM
I="MODN";J=1;"EDDG"
PSEV=UNION("EDDG" PHI)
*JZ(TT1,4102)
PATE=PROD(DEMP,PSEV)
*LABEL 4102
*JZ(TT2,4103)
PATE=PROD(-1. KGP,PSEV)
*LABEL 4103
PNIN=SUM(PATE,-1. PTRI)
*JZ(TT1,4600)
P LOAD "IDV"=SUM(PNIN,PTWO)
*LABEL 4600
*JZ(TT2,4601)
P LUAO "IDV"=SUM(PNIN,-1. PTWO)
*LABEL 4601
INIDV=IDV
DEFINE P1=P LOAD "IDV"
*JNZ,-1(NIDV,2005)
TMP1=UNION(1.0 P1)
*JUMP 2007
*LABEL 2005
SUM1=UNION(TMP1,P1)
TMP1=UNION(1.0 SUM1)
*LABEL 2007
*JZ(TT2,4110)
*XQT AUS
*LABEL 4110
*JUMP 2006
*LABEL 2001
APPL FORC 3 1=UNION(1.0 TMP1)
*JGZ,-1(NM1,2013)
*LABEL 2013
!JDF=DS,2,1,1(1,JDF1,BTAB,1,8)
DEFINE XX1=PREV "NAM8" 1 1 "MODN",MODN"
Z=MM1(XX1)
!Z1=DS,6,1,1(1,Z,AUS,MASK,MASK)
!Z4=Z1/"JDF"+1
TABLE(NI=1,NJ=1);Z24
J=1;"Z4"
ZZ5=IFIX(ZZ4)
!NJNT=DS,1,1,1(1,ZZ5,AUS,MASK,MASK)
!Z3=DS,1,1,1(1,JDF1,BTAB,1,8)
!NJN1=NJNT-1
!RRR="JDF"*NJN1
!TDEG=Z1-RRR
!NDEG=IFIX(TDEG)
!NDEG=DS,1,"NDEG",1(1,IN,EX,MASK,MASK)
*DCALL(BOND COND)
ZERO "NDEG";"NJNT"
*XQT RSI
RESET K=ASP
*XQT SSOL
RESET K=ASP
RESET SET=3
!NNDV=NDDV
*LABEL 2009
*JLZ,-1(NNDV,2008)
!IDV=NDDV-NNDV
*XQT AUS
DEFINE VDER=STAT DISP 3 1 "IDV","IDV"
DEFINE F=APPL FORC 3 1 "IDV","IDV"
F=UNION(F)
VDER=UNION(VDER)
*JZ(TT1,4105)

```

APPENDIX

```

TONE=PRDD(DEMP,VDER)
DEFINE DM=DMDV DIAG 0 "IDV"
*LABEL 4105
*JZ(TT2,4106)
TONE=PRDD(KGP,VDER)
DEFINE DM=DKG SPAR "NDF" "IDV"
*LABEL 4106
TTWO=XTYD(PHI,TONE)
TTRI=PRDD(DM,PHI)
TFOR=XTYD(PHI,0.5 TTRI)
*JZ(TT1,4108)
CEE=SUM(-1. TTWO,-1. TFOR)
*LABEL 4108
*JZ(TT2,4109)
CEE=SUM(TTWO,TFOR)
*LABEL 4109
*XQT U1
ICCE=DS,1,1,1(1,CEE,MASK,MASK,MASK)
*XQT AUS
TFIV=UNION("CCE" PHI)
DMOD "NAM9" "MODN" "IDV"=SUM(VDER,TFIV)
GGGG=PRDD(ASP,DMOD "NAM9" "MODN" "IDV")
GRGR=SUM(F,-1. GGGG)
*XQT AUS
*JUMP 2009
*LABEL 2008
INNM=NM
*JNZ,-1(NNM,2010)
*JUMP 2010
*LABEL 2000
INM=FREE()
INM1=FREE()
INNMD=FREE()
IMODN=FREE()
ISKP=FREE()
IBSE=FREE()
LAMM=FREE()
IDV=FREE()
INNDV=FREE()
ITT1=FREE()
ITT2=FREE()
IN5=FREE()
IZ1=FREE()
IZ4=FREE()
INJNT=FREE()
IZ3=FREE()
INJN1=FREE()
IRRR=FREE()
INDEG=FREE()
INNDV=FREE()
ICCE=FREE()
INNM=FREE()
*RETURN
*                                ENDNLM
$
$-----
$
$(BOND COND)                    ENDBCD
*XQT TAB
CON=1
ZERO 1,2,3,4,5,6;1,2
ZERO 1,2,4,5,6;15,16
*RETURN
*                                ENDBCD
$
$----- 18
$
$(NELS MET2)                    ENDNL2
$$          $$ CALCULATE DERIVATIVES OF
$$          $$ BUCKLING OR VIBRATION LOADS ANALYTICALLY.
*DCALL(BOND COND)
!JDF=DS,2,1,1(1,JDF1,BTAB,1,8)
INNMD=MODE
*XQT AUS
INNDV=NODV
*LABEL 4111
*JLZ,-1(NNDV,4110)
IIDV=NODV-NNDV

```


APPENDIX

```

TABLE(NJ=1,NI="MODE");DVAL "NAM9" 1 "IDV"
I=1;J=1;O.
*JUMP 4111
*LABEL 4110
!TT1=EQUAL(NAM9,MDE1)
!TT2=EQUAL(NAM9,MDE2)
*LABEL 2801
*JLZ,-1(NNMD,2800)$LUOP ON MODES
!NM=MODE-NNMD
!NM1=NM
!MODN=DS,1,"NM",1(1,MODE,NUM,MASK,MASK)
!ISKP=MODT-MODN
!IBSE=MODN-1
$      FIND EIGENVALUE
*XQT AUS
DEFINE L=PREV "NAM7" 1 1
TABLE(NI=1,NJ=1);LAM1 AUS 1 1
TRAN(SOURCE=L,SBASE="IBSE",ILIM=1,SSKIP="ISKP")
*XQT U1
!LAMM=DS,1,1,1(1,LAM1,AUS,1,1)
!LAMN=LAMM*.5
*XQT AUS
DEFINE PP=PREV "NAM8" 1 1 "MODN","MODN"
PHI=UNION(1.0 PP)
*JZ(TT1,4000)
LDEM DIAG 0 0=UNION("LAMM" DEMP)
ASP SPAR=SUM(KP,-1. LDEM)
*LABEL 4000
*JZ(TT2,4001)
LDEM DIAG 0 0=SUM("LAMN" KGP,"LAMN" KGP)
ASP SPAR=SUM(KP,LDEM)
*LABEL 4001
PONE=UNION("LAMM" PHI)
!NNDV=NDDV
*LABEL 2806
*JLZ,-1(NNDV,2801)
!IDV=NDDV-NNDV
!JDF=DS,2,1,1(1,JDF1,BTAB,1,8)
!NDF=JDF*JDF
*JZ(TT1,2803)
DEFINE DM=DMDV DIAG 0 "IDV"
*LABEL 2803
*JZ(TT2,2804)
DEFINE DM=DKG SPAR "NDF" "IDV"
*LABEL 2804
DEFINE DK=DKDV SPAR "NDF" "IDV"
PTWO=PKOD(DM,PONE)
PTRI=PRD(DK,PHI)
PFOR=XTYD(PHI,PTRI)
PFIV=XTYD(PHI,PTWO)
*JZ(TT1,4400)
PSIX=SUM(PFOR,-1. PFIV)
*LABEL 4400
*JZ(TT2,4401)
PSIX=SUM( PFOR, PFIV)
*LABEL 4401
EDER DIAG "MODN" "IDV"=UNION(1.0 PSIX)
!EDDG=DS,1,1,1(1,EDER,DIAG,"MODN","IDV")
TABLE,U(NJ=1,NI="MODE");DVAL "NAM9" 1 "IDV"
OPERATION=XSUM
I="MODN";J=1;"EDDG"
*JUMP 2806
*LABEL 2807
!NM=FREE()
!NM1=FREE()
!NNMD=FREE()
!MODN=FREE()
!IBSE=FREE()
!LAMM=FREE()
!IDV=FREE()
!NNDV=FREE()
!TT1=FREE()
!TT2=FREE()
!N5=FREE()
*JUMP 2801
*LABEL 2800
*LABEL 2807
*RETURN
*
```

ENDNL2

APPENDIX

```

$----- 19
$
$(MODL METH)          ENDMMD
$$          $$ CALCULATES DERIVATIVES OF VIB. OR BUCK. MODES
$$          $$ USING THE MODAL METHOD.

```

This runstream has been deleted.

```

$----- 20
$
$(PRT SUB)          ENDPSB
*XQT DCU
*JZ(DISP,2900)
IA=NLST
*LABEL 5666
*JLZ,-1(A,5667)
INNST=NLST-A
  PRINT 1 DUDV CAPV "NNST" 1
*JUMP 5666
*LABEL 5667
*LABEL 2900
*JZ(STRD,2901)
*DCALL(PRI DSTS)
*LABEL 2901
*JZ(VIBD,2902)
INAM9='VIBR
*DCALL(PRI DMOD)
*LABEL 2902
*JZ(BUCD,2903)
INAM9='BUCK
*DCALL(PRI DMOD)
*LABEL 2903
*RETURN
*          ENDPSB
$-----
$
$(PRI DSTS)          ENDPST
$$          $$ PRINT STRESS DERIVATIVES
*JZ(DE21,3104)
IE=DE21
INAM5='E21
*DCALL(PRI DSIG)
*LABEL 3104
*JZ(DE23,3105)
IE=DE23
INAM5='E23
*DCALL(PRI DSIG)
*LABEL 3105
*JZ(DE31,3106)
IE=DE31
INAM5='E31
*DCALL(PRI DSIG)
*LABEL 3106
*JZ(DE32,3107)
IE=DE32
INAM5='E32
*DCALL(PRI DSIG)
*LABEL 3107
*JZ(DE33,3108)
IE=DE33
INAM5='E33
*DCALL(PRI DSIG)
*LABEL 3108
*JZ(DE41,3109)
IE=DE41
INAM5='E41
*DCALL(PRI DSIG)
*LABEL 3109
*JZ(DE42,3110)
IE=DE42
INAM5='E42
*DCALL(PRI DSIG)
*LABEL 3110
*JZ(DE43,3111)
IE=DE43
INAM5='E43

```

APPENDIX

```

*DCALL(PRIN DSIG)
*LABEL 3111
*JZ(DE44,3112)
!IE=DE44
!NAM5='DE44
*DCALL(PRIN DSIG)
*LABEL 3112
!IE1=FREE()
!NIE1=FREE()
!N1=FREE()
!IE=FREE()
!NAM5=FREE()
*RETURN
*
ENDPST
$
$-----
$
*(PRIN DSIG)          ENDPDS
!IE2=IE
*LABEL 3113
*JLZ,-1(!IE,3114)$ LOOP OVER GROUP NUMBER
!IV=IE2-IE$GROUP NUMBER
!IE1=NLST$ LOOP OVER LOAD SETS
*LABEL 3102
*JLZ,-1(!IE1,3103)$LOAD SET NUMBER
!NIE1=NLST-IE1
PRINT 1 DSIG "NAM5" "NIE1" "IV"
*JUMP 3102
*LABEL 3103
*JUMP 3113
*LABEL 3114
*RETURN
*
ENDPDS
$
$-----
$
*(PRIN DMDD)          ENDPMD
$$          $$ PRINT VIBRATION AND OR BUCKLING DERIVATIVES
!NI1=NDV
*LABEL 3120
*JLZ,-1(!NI1,3121)$LOOP OVER DESIGN VARIABLES
!IV=NDV-NI1$DESIGN VARIABLE NUMBER
!NMD=MODE
*LABEL 3122
!TT1=EQUAL(NAM9,MDE1)
!TT2=EQUAL(NAM9,MDE2)
*JZ(TT1,3125)
!N1=VIBD
*JZ,-2(N1,3123)
*LABEL 3125
*JZ(TT2,3126)
!N1=BUCD
*JZ,-2(N1,3123)
*LABEL 3126
*JLZ,-1(!NMD,3123)$LOOP OVER MODES
!NM=MODE-NMD
!MODN=DS,1,"NM",1(1,MODE,NUM,MASK,MASK)$MODE NUMBER
PRINT 1 DMDD "NAM9" "MODN" "IV"
*JUMP 3126
*LABEL 3123
!TT1=EQUAL(NAM9,MDE1)
!TT2=EQUAL(NAM9,MDE2)
*JZ(TT1,3127)
!N1=VIBD
*JZ,-4(N1,3120)
*LABEL 3127
*JZ(TT2,3128)
!N1=BUCD
*JZ,-4(N1,3120)
*LABEL 3128
PRINT 1 DVAL "NAM9" 1 "IV"
*JUMP 3120
*LABEL 3121
!NI1=FREE()
!IV=FREE()
!TT1=FREE()
!TT2=FREE()
!NN1=FREE()
!N1=FREE()

```

APPENDIX

```
INM=FREE()  
IMODN=FREE()  
*RETURN  
*  
*  
*  
*  
*----- 21  
*  
*DCALL(DRIV GRD4)  
*XQT EXIT
```

REFERENCES

1. Sobieszczanski-Sobieski, Jaroslaw: Structural Optimization: Challenges and Opportunities. Paper presented at International Conference on Modern Vehicle Design Analysis (London, England), June 1983.
2. Haftka, R. T.; and Prasad, B.: Programs for Analysis and Resizing of Complex Structures. Trends in Computerized Structural Analysis and Synthesis, Ahmed K. Noor and Harvey G. McComb, Jr., eds., Pergamon Press Ltd., c.1978, pp. 323-330. (Also available in Comput. & Struct., vol. 10, no. 1/2, Apr. 1979, pp. 323-330.)
3. Schmit, L. A.; and Miura, H.: A New Structural Analysis/Synthesis Capability - ACCESS 1. AIAA J., vol. 14, no. 5, May 1976, pp. 661-671.
4. Giles, Gary L.; and Rogers, James L., Jr.: Implementation of Structural Response Sensitivity Calculations in a Large-Scale Finite-Element Analysis System. A Collection of Technical Papers, Part 2: Structural Dynamics and Engineering - AIAA/ASME/ASCE/AHS 23rd Structures, Structural Dynamics and Materials Conference, May 1982, pp. 348-367. (Available as AIAA-82-0714.)
5. Adelberg, Michael L.; DeVries, Richard I.; Moschovidis, Zissis A.; and Razgunas, Leo: OPUS: A Programming System Approach to Structural Optimization. 4th International Conference on Vehicle Structural Mechanics - p. 99, Soc. Automot. Eng., Inc., Nov. 1981, pp. 151-159. (Available as SAE Preprint 811316.)
6. Fox, Richard L.: Optimization Methods for Engineering Design. Addison-Wesley Pub. Co., Inc., c.1971.
7. Arora, Jasbir S.; and Haug, Edward J.: Methods of Design Sensitivity Analysis in Structural Optimization. AIAA J., vol. 17, no. 9, Sept. 1979, pp. 970-974.
8. Nelson, Richard B.: Simplified Calculation of Eigenvector Derivatives. AIAA J., vol. 14, no. 9, Sept. 1976, pp. 1201-1205.
9. Cardani, C.; and Mantegazza, P.: Calculation of Eigenvalue and Eigenvector Derivatives for Algebraic Flutter and Divergence Eigenproblems. AIAA J., vol. 17, no. 4, Apr. 1979, pp. 408-412.
10. Morris, A. J., ed: Foundations of Structural Optimization: A Unified Approach. John Wiley & Sons, c.1982.
11. Prasad, B.; and Emerson, J. F.: A General Capability of Design Sensitivity for Finite Element Systems. A Collection of Technical Papers, Part 2: Structural Dynamics and Design Engineering - AIAA/ASME/ASCE/AHS 23rd Structures, Structural Dynamics and Materials Conference, May 1982, pp. 175-186. (Available as AIAA-82-0680.)
12. Whetstone, W. D.: EISI-EAL Engineering Analysis Language Reference Manual - EISI-EAL System Level 2091. Engineering Information Systems, Inc., July 1983.
13. Whetstone, W. D.: SPAR Structural Analysis System Reference Manual - System Level 13A. Volume I: Program Execution. NASA CR-158970-1, 1978.

14. Giles, Gary L.; and Haftka, Raphael T.: SPAR Data Handling Utilities. NASA TM-78701, 1978.
15. Cunningham, Sally W.: SPAR Data Set Contents. NASA TM-83181, 1981.
16. Prasad, Birendra: Automated Design of Structures Subject to Stress, Displacement, and Temperature Constraints. Ph. D. Thesis, Illinois Inst. of Technol., Dec. 1977.
17. Schmit, Lucien A., Jr.; and Miura, Hirokazu: Approximation Concepts for Efficient Structural Synthesis. NASA CR-2552, 1976.
18. Rizzi, Paulo: Optimization of Multi-Constrained Structures Based on Optimality Criteria. Proceedings - AIAA/ASME/SAE 17th Structures, Structural Dynamics, and Materials Conference, May 1976, pp. 448-462.
19. Haug, Edward J.; and Arora, Jasbir S.: Applied Optimal Design - Mechanical and Structural Systems. John Wiley & Sons, Inc., c.1979.
20. Hauser, Charles C., III; Musgrove, Max D.; and Reed, James M.: Optimization of Boeing IUS ASE Design Using Sensitivity Analysis. A Collection of Technical Papers, Part 1 - AIAA/ASME/ASCE/AHS 22nd Structures, Structural Dynamics & Materials Conference, AIAA, Apr. 1981, pp. 121-133. (Available as AIAA-81-0516.)
21. Sobieszczanski-Sobieski, Jaroslaw; and Rogers, James L., Jr.: A Programing System for Research and Applications in Structural Optimization. International Symposium on Optimum Structural Design - The 11th ONR Naval Structural Mechanics Symposium, U.S. Office of Naval Research and Univ. of Arizona, Oct. 1981, pp. 11-9 - 11-21.
22. Garrett, L. Bernard: Interactive Design and Analysis of Future Large Spacecraft Concepts. NASA TP-1937, 1981.
23. Haftka, Raphael T.; and Adelman, Howard M.: An Analytical Investigation of Shape Control of Large Space Structures by Applied Temperatures. NASA TM-85649, 1983.

TABLE 1.- FUNCTIONS OF EAL PROCESSORS

Processor name	Function
TAB	Creates data sets containing tables of joint locations, section properties, material constants, and so forth
ELD	Defines the finite-element connections in model
E	Generates sets of information for each element, including connected joint numbers, geometrical data, material, and section property data
EKS	Adds the stiffness and stress matrices for each element to the set of information produced by the E processor
SEQ	Determines joint sequences, i.e., equation numbering sequences to be used in sparse matrix solution methods
TOPO	Analyzes element interconnections and topology and creates data sets used to assemble and factor the system mass and stiffness matrices
K	Assembles the unconstrained system stiffness matrix in a sparse matrix format
M	Assembles the unconstrained system mass matrix in a sparse matrix format
KG	Assembles the unconstrained system initial-stress (geometric stiffness) matrix in a sparse matrix format
INV	Factors the assembled system matrices
RSI	Similar to INV
DRSI	Similar to RSI; factors double precision SPAR format matrices
EQNF	Computes equivalent joint loading associated with thermal, dislocational, and pressure loading
SSOL	Computes displacements and reactions due to applied loading at the joints
TAN	Similar to TOPO
LSK	Forms partial or complete system stiffness (K) and damping (D) matrices in a sparse matrix form called LS-format
LSU, } RMK }	Translates arbitrary source K and M data into LS- or SPAR-format and transforms SPAR- and LS-format matrices
GSF	Generates element stresses and internal loads
PSF	Prints the information generated by the GSF processor
ES	Analyzes element interior state, given joint displacements, and initial strains, if present; creates and stores stresses and internal load data

TABLE 1.- Concluded

Processor name	Function
EIG	Solves linear vibration and bifurcation buckling eigenproblems
E4	Similar to EIG
SYN	Produces mass and stiffness matrices for systems comprised of interconnected substructures
STRP	Computes eigenvalues and eigenvectors of substructured systems
SSBT	Back-transforms synthesized system results into individual substructure terms used in conjunction with SYN and STRP
AUS	Performs matrix arithmetic functions and is used in construction, editing, and modification of data sets
DCU	Performs data management functions including display of table of contents, data transfer between libraries, changing data set names, printing data sets, and transferring data between libraries and sequential files
VPRT	Performs editing and printing of data sets which are in the form of vectors on the data libraries
PLTA, PLTB, PXY }	Produce graphic displays of finite-element models and computed results such as vibration and buckling modes, stresses, and response histories
PR	Generates reports of dynamic response analysis
DR	Computes linear transient modal response and back-transforms to determine any required system response details and maximum-minimum and time-of-occurrence data
U1	Creates, edits, and manipulates runstreams and permits direct table input
U3/RP2	Produces tabular multipage reports, using formats, headings, and footnotes prescribed by the user at execution time
U4/VU	Enables vector arithmetic functions
FSM	Creates SPAR-format matrices for compressible fluid elements
PS	Prints SPAR-format matrices and factored system matrices produced by RSI or DRSI
EI1	Utility processor which operates on element state information
PRTE	Prints contents of the element state

TABLE 2.- EAL STRUCTURAL ELEMENT DESCRIPTIONS

Name	Description
E21	Beam element
E23	Rod element
E31, E41	Triangular and quadrilateral membrane elements, respectively
E32, E42	Triangular and quadrilateral uncoupled (bending only) plate elements, respectively
E33, E43	Triangular and quadrilateral coupled (membrane and bending) plate elements, respectively
E44	Shear panel element

TABLE 3.-- NODAL COORDINATES FOR SWEPT WING

Node	x, in.	y, in.	z, in.	Node	x, in.	y, in.	z, in.
1	0	300.00	10.00	45	0	300.00	0
2	0	250.00	15.00	46	0	250.00	
3	0	185.00	13.00	47	0	185.00	
4	0	100.00	5.00	48	0	100.00	
5	100.0	258.30	8.58	49	100.0	258.30	
6	100.0	214.20	12.83	50	100.0	214.20	
7	100.0	157.20	11.08	51	100.0	157.20	
8	100.0	85.33	4.33	52	100.0	85.33	
9	190.0	220.80	7.31	53	190.0	220.80	
10	190.0	181.90	10.88	54	190.0	181.90	
11	190.0	132.10	9.36	55	190.0	132.10	
12	190.0	68.33	3.73	56	190.0	68.33	
13	260.0	191.70	6.32	57	260.0	191.70	
14	260.0	156.80	9.37	58	260.0	156.80	
15	260.0	112.60	8.02	59	260.0	112.60	
16	260.0	56.67	3.27	60	260.0	56.67	
17	325.0	164.60	5.40	61	325.0	164.60	
18	325.0	133.50	7.96	62	325.0	133.50	
19	325.0	94.54	6.77	63	325.0	94.54	
20	325.0	45.83	2.83	64	325.0	45.83	
21	385.0	139.00	4.55	65	385.0	139.00	
22	385.0	112.00	6.66	66	385.0	112.00	
23	385.0	77.84	5.62	67	385.0	77.84	
24	385.0	35.83	2.43	68	385.0	35.83	
25	440.0	116.70	3.77	69	440.0	116.70	
26	440.0	92.33	5.47	70	440.0	92.33	
27	440.0	62.53	4.57	71	440.0	62.53	
28	440.0	26.67	2.07	72	440.0	26.67	
29	490.0	95.83	3.06	73	490.0	95.83	
30	490.0	74.42	4.38	74	490.0	74.42	
31	490.0	48.62	3.61	75	490.0	48.62	
32	490.0	18.33	1.73	76	490.0	18.33	
33	535.0	77.08	2.42	77	535.0	77.08	
34	535.0	58.29	3.41	78	535.0	58.29	
35	535.0	36.09	2.75	79	535.0	36.09	
36	535.0	10.83	1.43	80	535.0	10.83	
37	570.0	62.50	1.93	81	570.0	62.50	
38	570.0	45.75	2.65	82	570.0	45.75	
39	570.0	26.35	2.08	83	570.0	26.35	
40	570.0	5.00	1.20	84	570.0	5.00	
41	600.0	50.00	1.50	85	600.0	50.00	
42	600.0	35.00	2.00	86	600.0	35.00	
43	600.0	18.00	1.50	87	600.0	18.00	
44	600.0	0	1.00	88	600.0	0	

TABLE 4.- DESIGN VARIABLE GROUPS FOR SWEEP WING

Design-variable type	Design-variable number	Element numbers
Areas of E23 elements	1	1
	2	2
	3	3
	4	4
	5	5, 6
	6	7, 8
	7	9, 10
	8	11
	9	12
	10	13
	11	14
	12	15, 16
	13	17, 18
	14	19, 20
Thickness of E31 elements	15	1-6
	16	7-12
	17	13-18
	18	19-24
	19	25-36
	20	37-48
	21	49-60
Thickness of E44 elements	22	1-4
	23	5-10
	24	11-14
	25	15-20
	26	21-24
	27	25-30
	28	31-34
	29	35-40
	30	41-49
	31	50-58
	32	59-70

TABLE 5.- LOAD DATA FOR SWEPT WING

$$[f_x = 0 \text{ and } f_y = 0]$$

Node	$f_{z'}$ lb	Node	$f_{z'}$ lb	Node	$f_{z'}$ lb
Load condition 1					
5	1282.0	19	1453.0	33	206.0
6	2581.0	20	1057.0	34	431.0
7	3398.0	21	459.0	35	563.0
8	2380.0	22	958.0	36	383.0
9	978.0	23	1251.0	37	144.0
10	2013.0	24	852.0	38	302.0
11	2593.0	25	362.0	39	395.0
12	1764.0	26	756.0	40	269.0
13	727.0	27	986.0	41	62.0
14	1386.0	28	671.0	42	129.0
15	1906.0	29	282.0	43	169.0
16	1297.0	30	589.0	44	116.0
17	570.0	31	768.0		
18	1190.0	32	522.0		
Load condition 2					
5	2361.0	19	1025.0	33	402.0
6	3876.0	20	355.0	34	646.0
7	2308.0	21	843.0	35	398.0
8	793.0	22	1374.0	36	154.0
9	1772.0	23	825.0	37	311.0
10	2895.0	24	284.0	38	482.0
11	1705.0	25	665.0	39	306.0
12	582.0	26	1092.0	40	135.0
13	1310.0	27	651.0	41	133.0
14	2135.0	28	224.0	42	206.0
15	1258.0	29	518.0	43	131.0
16	433.0	30	851.0	44	58.0
17	1047.0	31	508.0		
18	1719.0	32	175.0		

TABLE 6.- SUMMARY OF SENSITIVITY ANALYSIS FOR SWEPT WING

[88 nodes; 194 elements; 2 load conditions; 32 design
variables; 164 degrees of freedom]

(a) Convergence for displacement derivatives

Change, percent, in design variable	Error, ^a percent for -	
	Finite-difference method	Semianalytical method
1.000	0.900	0
.100	.090	0
.010	.008	
.001	.030	

^aRelative to analytical method.

(b) Solution time for displacement derivatives

Method	Solution time, sec
Finite difference	700
Semianalytical without LSK processor	419
Semianalytical with LSK processor	161
Analytical	135

TABLE 7.- SUMMARY OF SENSITIVITY ANALYSIS FOR BOX BEAM

28 nodes; 77 elements; 1 load condition; 9 independent design variables; 10 dependent design variables; 122 degrees of freedom

(a) Solution time comparison

Method	Solution time, sec, for displacement derivatives
Finite difference	133
Semianalytical with LSK processor	63

(b) Typical values of derivatives

Design variable	$\partial u_{tip} / \partial V$, in/in.
Area, E23	Negligible
Thickness:	
E31	-1.2
E41	-1.2
E44	-1.5
E32	Negligible
E33	-8.4
E42	Negligible
E43	-8.4
Radius, E21	-1.2

TABLE 8.- NODAL COORDINATES FOR STIFFENED CYLINDER

Node	x, in.	y, in.	z, in.	Node	x, in.	y, in.	z, in.
1	157.50	0	0 ↓	41	157.50	0	157.50 ↓
2	145.50	60.26		42	-145.50	-60.26	
3	111.40	111.40		43	-111.40	-111.40	
4	60.26	145.50		44	-60.26	-145.50	
5	0	157.50		45	0	-145.50	
6	-60.26	145.50		46	60.26	-145.50	
7	-111.40	111.40		47	111.40	-111.40	
8	-145.50	60.26		48	145.50	-60.26	
9	-157.50	0		236.20 ↓	49	157.50	0
10	-145.50	-60.26			50	145.50	60.26
11	-111.40	-111.40			51	111.40	-111.40
12	-60.26	-145.50			52	-60.26	145.50
13	0	-157.50			53	0	157.50
14	60.26	-145.50			54	-60.26	145.50
15	111.40	-111.40			55	-111.40	111.40
16	145.50	-60.26			56	-145.50	60.26
17	157.50	0	78.74 ↓		57	-157.50	0
18	145.50	60.26			58	-145.50	-60.26
19	111.40	111.40			59	-111.40	-111.40
20	60.26	145.50			60	-60.26	-145.50
21	0	157.50			61	0	-157.50
22	-60.26	145.50			62	60.26	-145.50
23	-111.40	111.40			63	111.40	-111.40
24	-145.50	60.26			64	145.50	60.26
25	-157.50	0		315.00 ↓	65	157.50	0
26	-145.50	-60.26			66	145.50	60.26
27	-111.40	-111.40			67	111.40	111.40
28	-60.26	-145.50			68	60.26	145.50
29	0	-157.50			69	0	157.50
30	60.26	-145.50			70	-60.26	145.50
31	111.40	-114.40			71	-111.40	111.40
32	145.40	-60.26			72	-145.50	60.26
33	157.50	0	157.50 ↓		73	157.50	0
34	145.50	60.26			74	-145.50	-60.26
35	111.40	111.40			75	-111.40	-111.40
36	60.26	145.50			76	-60.26	-145.50
37	0	157.50			77	0	-157.50
38	-60.26	145.50			78	60.26	-145.50
39	-111.40	111.40			79	111.40	-111.40
40	-145.50	60.26			80	145.50	-60.26

TABLE 9.- SUMMARY OF SENSITIVITY ANALYSIS FOR STIFFENED CYLINDER WITH A CUTOUT

[80 nodes; 190 elements; 1 load condition; 3 independent design variables; 5 dependent design variables; 352 degrees of freedom]

Method	Solution time, sec, for derivatives of -			
	Displacements	Stresses	Frequencies and mode shapes	Buckling loads and mode shapes
Finite difference	102	122	467	349
Semianalytical without LSK processor	76	95		
Semianalytical with LSK processor	73	92		
Analytical			201	220

TABLE 10.- NODAL COORDINATES FOR ANTENNA TRUSS

Node	x, in.	y, in.	z, in.	Node	x, in.	y, in.	z, in.
1	-625.40	1083.00	116.80	56	312.70	-1083.00	94.25
2	-781.80	812.50	94.25	57	1251.00	0	116.80
3	-938.20	541.60	86.73	58	1095.00	-270.80	94.25
4	-1095.00	270.80	94.25	59	938.10	-541.60	86.73
5	-1251.00	0	116.80	60	781.80	-812.50	94.25
6	-312.70	1083.00	94.25	61	625.40	-1083.00	116.80
7	-469.10	812.50	64.16	62	-629.10	910.20	41.28
8	-625.40	541.60	49.12	63	-786.90	636.70	25.30
9	-781.80	270.80	49.12	64	-944.80	363.10	25.30
10	-938.20	0	64.16	65	-1103.00	89.70	41.28
11	-1095.00	-270.80	94.25	66	-314.60	910.70	17.58
12	0	1083.00	86.73	67	-472.30	637.30	-6.34
13	-156.40	812.50	49.12	68	-630.10	363.80	-14.31
14	-312.70	541.60	26.55	69	-788.00	90.37	-6.34
15	-469.10	270.80	19.03	70	-946.00	-182.90	17.58
16	-625.40	0	26.55	71	0	910.80	9.68
17	-781.80	-270.80	49.12	72	-157.40	637.50	-22.14
18	-938.20	-541.60	86.73	73	-315.10	364.20	-38.05
19	312.70	1083.00	94.25	74	-472.90	90.81	-38.05
20	156.40	812.50	49.12	75	-630.80	-182.40	-22.14
21	0	541.60	19.03	76	-788.80	-455.40	9.68
22	-156.40	270.80	3.98	77	314.60	910.70	17.58
23	-312.70	0	3.98	78	157.40	637.50	-22.14
24	-469.10	-270.80	19.03	79	0	364.30	-45.96
25	-625.40	-541.60	49.12	80	-157.70	91.03	-53.90
26	-781.80	-812.50	94.25	81	-315.50	-182.10	-45.96
27	625.40	1083.00	116.80	82	-473.40	-455.10	-22.14
28	469.10	812.50	64.16	83	-631.40	-727.80	17.58
29	312.70	541.60	26.55	84	629.10	910.20	41.28
30	156.40	270.80	3.98	85	472.30	637.30	-6.34
31	0	0	-3.54	86	315.10	364.20	-38.05
32	-156.40	-270.80	3.98	87	157.70	91.03	-53.90
33	-312.70	-541.60	26.55	88	0	-182.10	-53.90
34	-469.10	-812.50	64.16	89	-157.80	-455.00	-38.05
35	-625.40	-1083.00	116.80	90	-315.70	-727.60	-6.34
36	781.80	812.50	94.25	91	-473.70	-999.90	41.28
37	625.40	541.60	49.12	92	786.90	636.70	25.30
38	469.10	270.80	19.03	93	630.10	363.80	-14.31
39	312.70	0	3.98	94	472.90	90.81	-38.05
40	156.40	-270.80	3.98	95	315.50	-182.10	-45.96
41	0	-541.60	19.03	96	157.80	-455.00	-38.05
42	-156.40	-812.50	49.12	97	0	-727.60	-14.31
43	-312.70	-1083.00	94.25	98	-157.90	-999.70	25.30
44	938.10	541.60	86.73	99	944.80	363.10	25.30
45	781.80	270.80	49.12	100	788.00	90.37	-6.34
46	625.40	0	26.55	101	630.80	-182.40	-22.14
47	469.10	-270.80	19.03	102	473.40	-455.10	-22.14
48	312.70	-541.60	26.55	103	315.70	-727.60	-6.33
49	156.40	-812.50	49.12	104	157.90	-999.80	25.30
50	0	-1083.00	86.73	105	1103.00	89.70	41.28
51	1095.00	270.80	94.25	106	946.00	-182.90	17.58
52	938.20	0	64.16	107	788.80	-455.40	9.68
53	781.80	-270.80	49.12	108	631.40	-727.80	17.58
54	625.40	-541.60	49.12	109	473.70	-999.90	41.28
55	469.10	-812.50	64.16				

TABLE 11.- THERMAL LOADING ON RADIOMETER ANTENNA

[Values are given relative to 80°F reference temperature]

Node	Temperature, °F	Node	Temperature, °F	Node	Temperature, °F
1	5.1	38	15.9	75	18.1
2	8.9	39	15.8	76	14.3
3	8.4	40	15.3	77	21.5
4	9.8	41	19.6	78	14.0
5	21.8	42	18.9	79	15.2
6	17.2	43	33.5	80	16.8
7	11.9	44	8.4	81	18.0
8	13.1	45	14.4	82	19.4
9	14.4	46	15.5	83	15.9
10	17.1	47	18.5	84	13.0
11	16.6	48	18.0	85	14.1
12	17.1	49	18.9	86	15.3
13	13.4	50	33.4	87	16.8
14	12.9	51	9.8	88	18.2
15	15.9	52	15.5	89	19.3
16	15.6	53	18.5	90	20.6
17	18.5	54	19.7	91	20.3
18	18.2	55	20.8	92	10.6
19	17.2	56	33.4	93	15.3
20	13.4	57	21.8	94	16.7
21	14.5	58	16.6	95	18.0
22	14.4	59	18.2	96	19.3
23	14.1	60	19.6	97	20.3
24	18.5	61	25.5	98	27.0
25	19.7	62	13.0	99	12.0
26	19.6	63	10.6	100	16.7
27	31.3	64	12.0	101	18.1
28	13.5	65	15.7	102	19.4
29	12.9	66	21.4	103	20.6
30	12.7	67	14.1	104	27.0
31	17.3	68	15.3	105	15.7
32	17.0	69	16.6	106	12.7
33	18.0	70	12.7	107	14.3
34	19.1	71	21.4	108	15.8
35	25.3	72	14.0	109	20.3
36	6.9	73	15.3		
37	13.1	74	16.7		

TABLE 12.- SUMMARY OF ANTENNA REFLECTOR
SENSITIVITY ANALYSIS

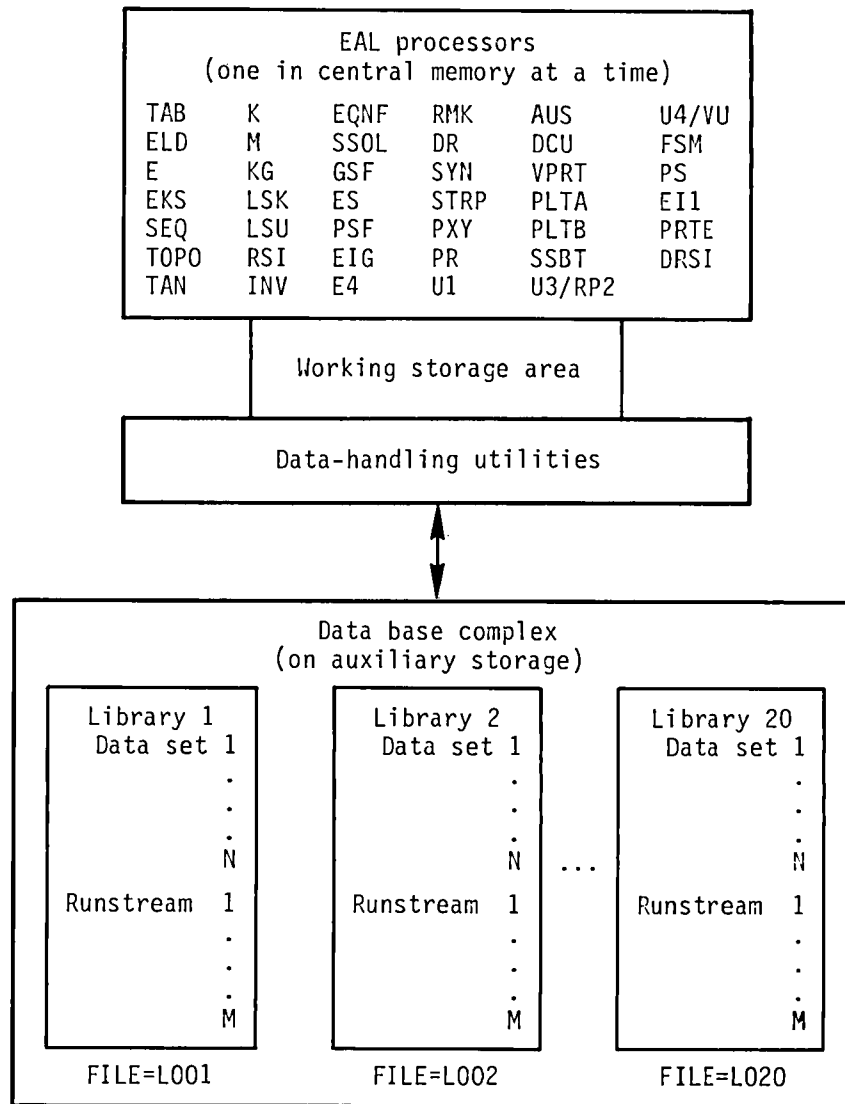
[109 nodes; 420 elements; 1 load condition;
3 design variables]

(a) Solution time comparison

Method	Solution time, sec, for derivatives of -	
	Displacement	Stress
Finite difference	155	160
Semianalytical	95	114

(b) Comparison of derivatives of
center deflection

Independent variable, cross-sectional area	Value of derivative of center deflection
Upper surface	2.4×10^{-4}
Diagonals	-8.3×10^{-5}
Lower surface	-1.8×10^{-4}



(a) System configuration.

```

*XQT TAB
  UPDATE=1
  (TAB DATA)
*CALL (1,CHNG DV)
*XQT ELD
  (ELD DATA)
*XQT K
*XQT INV

```

(b) Sample input file.

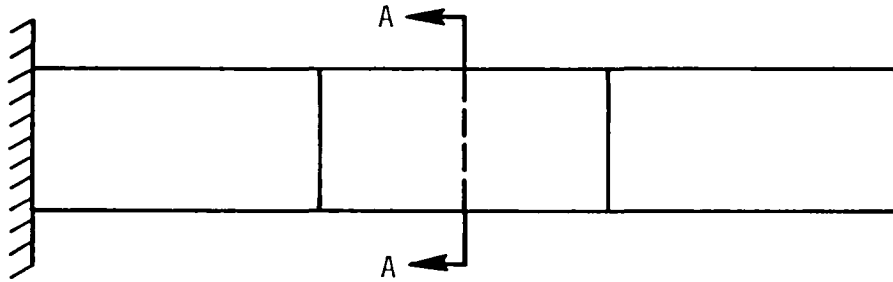
```

*(1,CHNG DV)  ENDCDV
!I=NUMEL
*LABEL 1
*JLZ,-1(NUMEL,2)$LOOP OVER ELEMENT NO.'S
!NI=I-NUMEL
$SELECT NEW VALUE OF DESIGN VARIABLE
!DV=DS,1,"NI",1(1,DV,DFN,MASK,MASK)
$UPDATE DESIGN VARIABLE TO NEW VALUE
$MEMBRANE THICKNESSES
SA
"NI"      "DV"
*JUMP     1
*LABEL    2
*RETURN
*ENDCDV

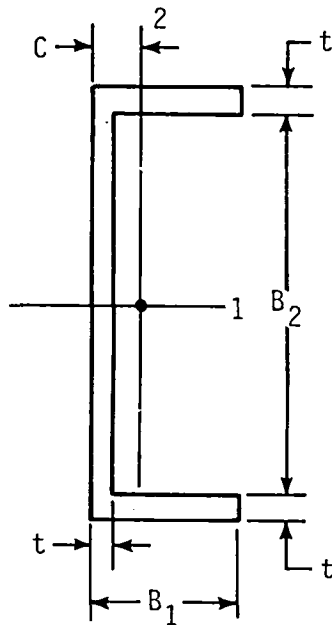
```

(c) Sample runstream data set.

Figure 1.- The EAL system terminology and configuration.



(a) Finite-element model of cantilevered channel-section beam.



(b) Section A-A showing cross-sectional geometry and structural variables.

Figure 2.- Cantilevered channel-section beam used to illustrate analytical derivative method.

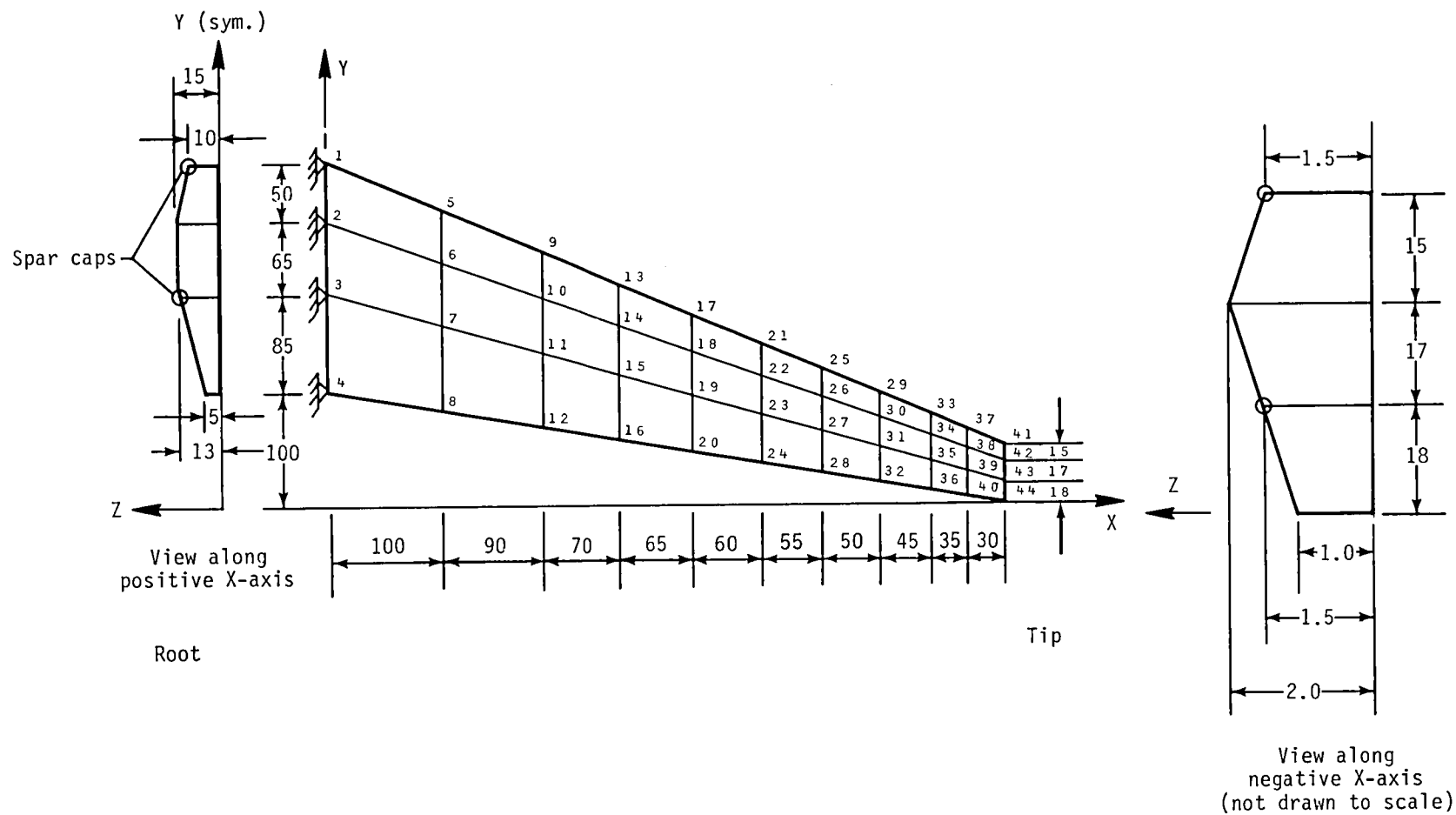


Figure 3.- Geometry and node numbering for swept wing. All dimensions are in inches.

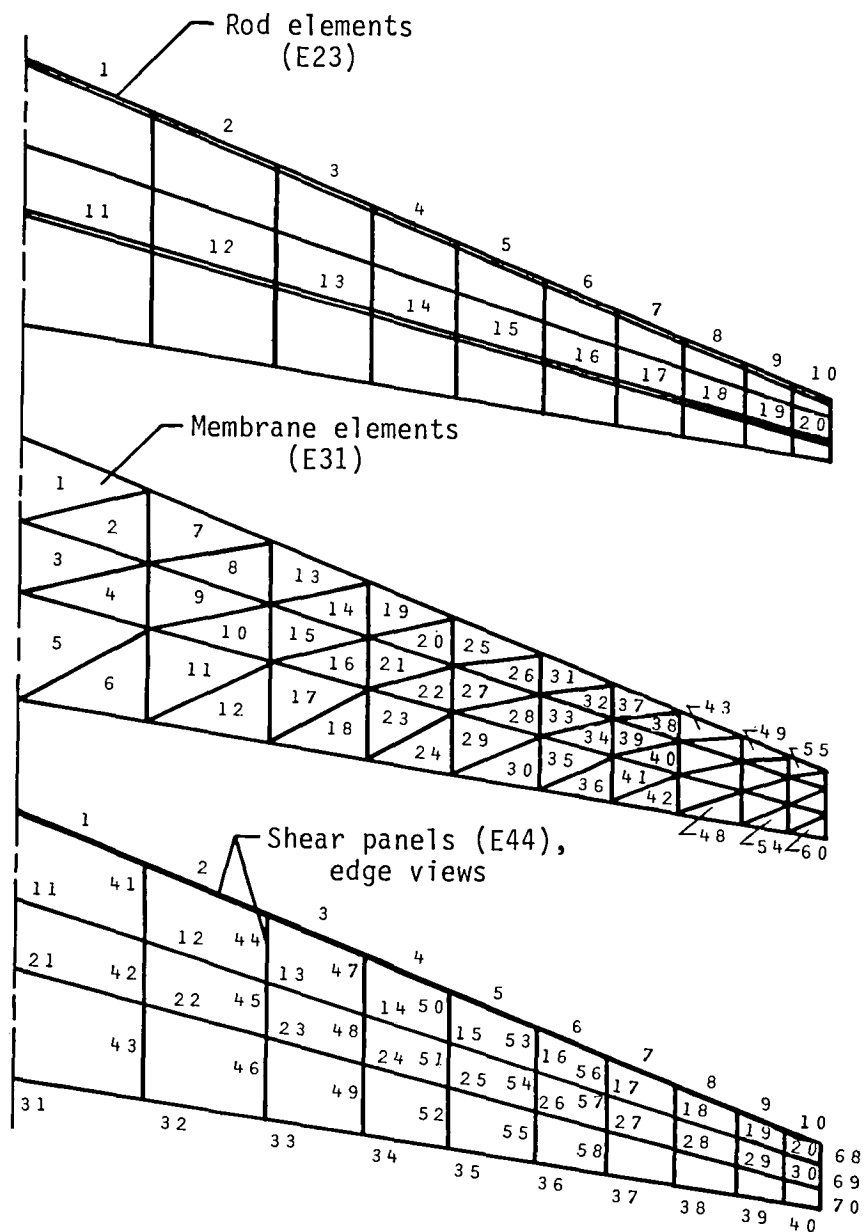


Figure 4.- Element numbering for swept wing.

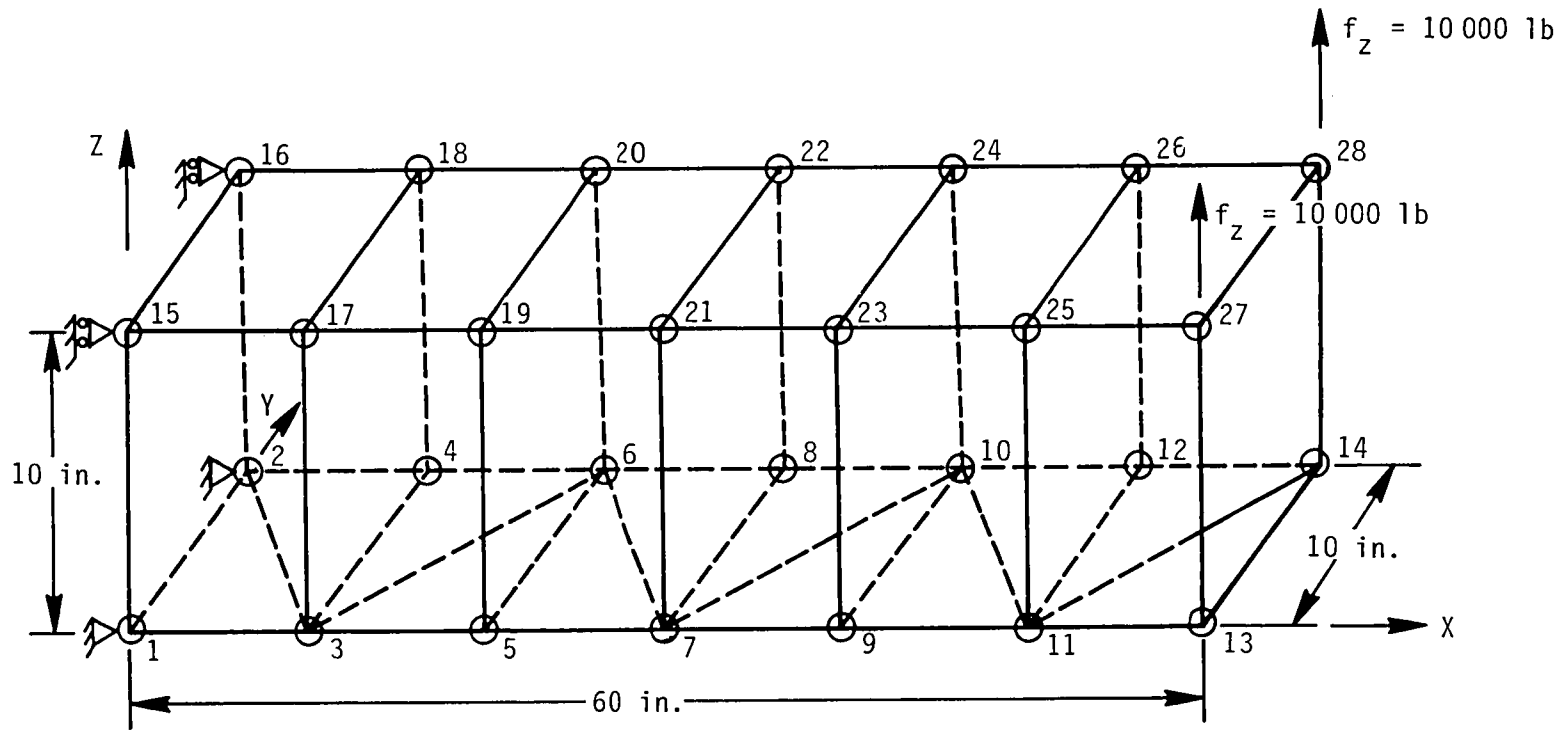


Figure 5.- Overall geometry and nodal numbering for box beam.

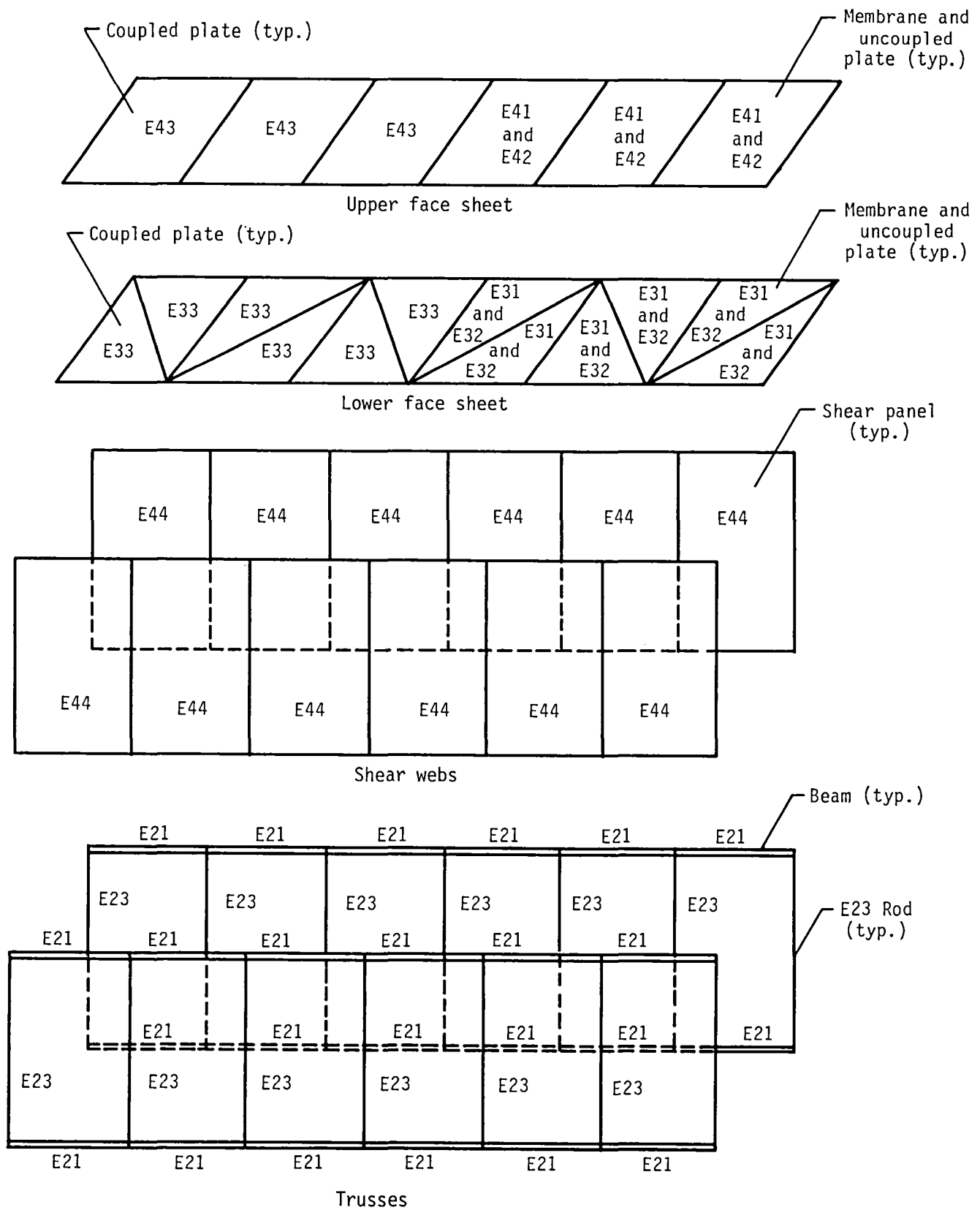


Figure 6.- Elements for box beam.

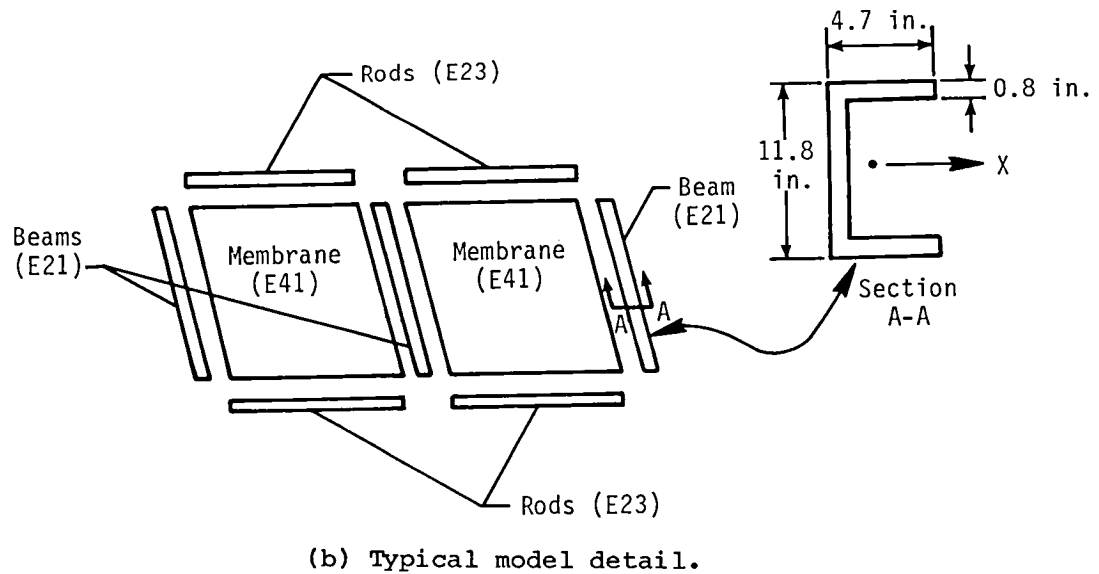
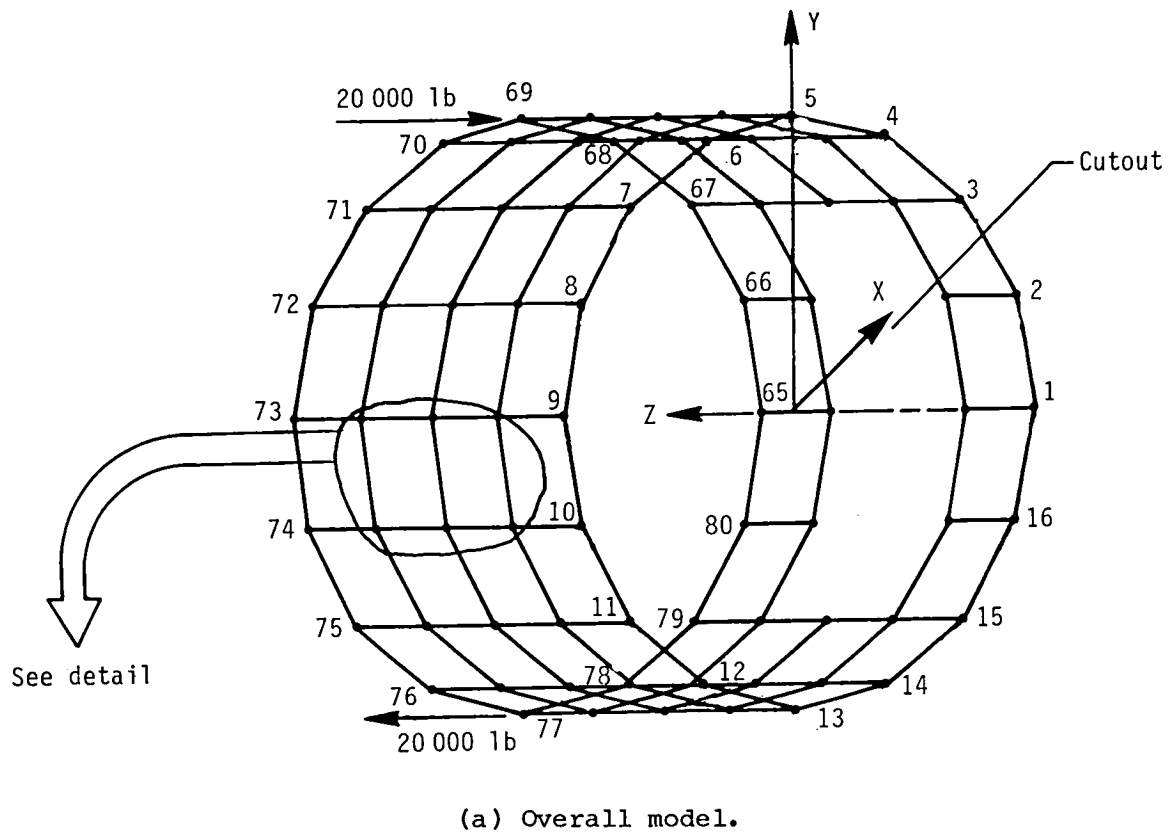
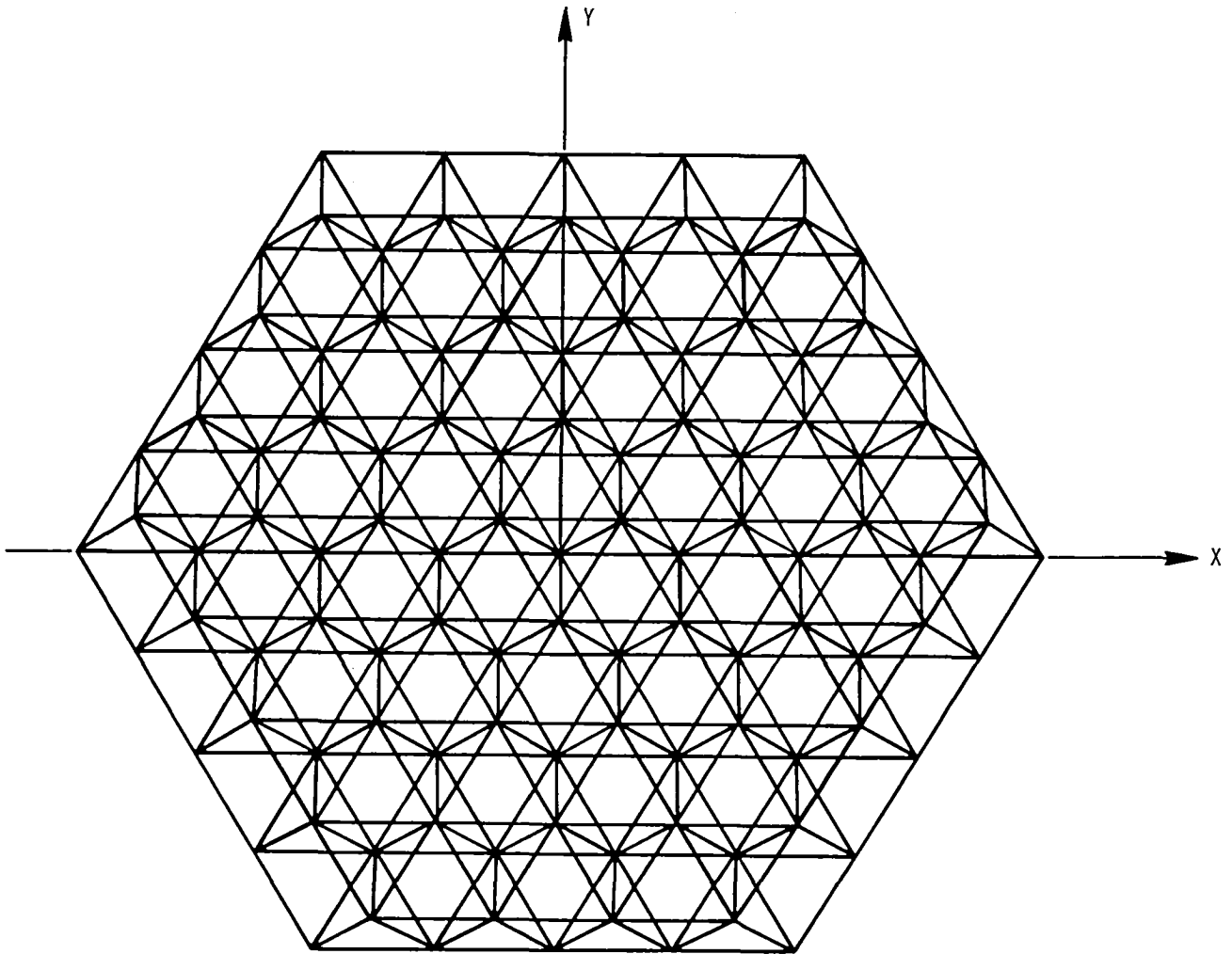
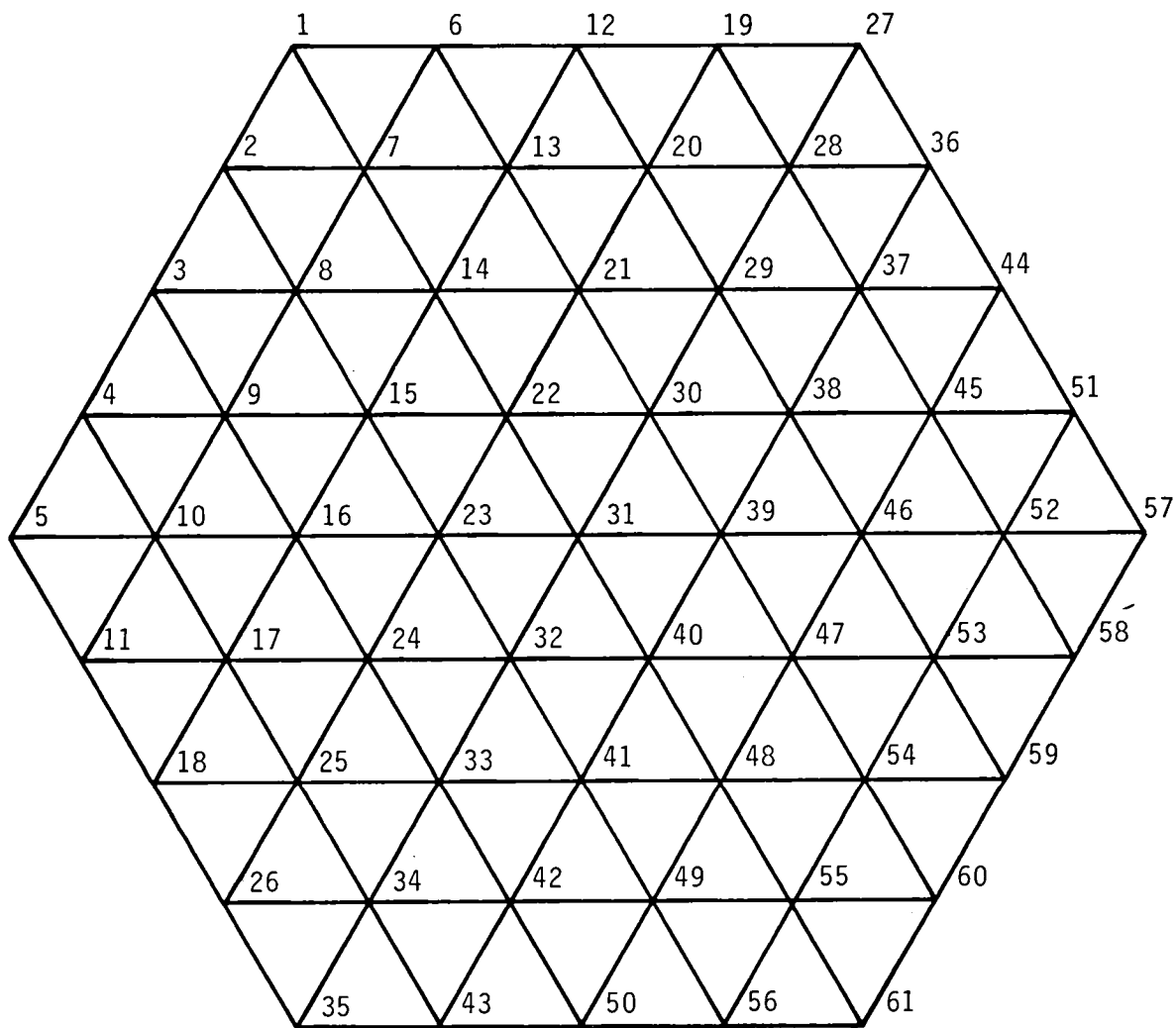


Figure 7.- Overall geometry and nodal numbering for stiffened cylinder with cutout.



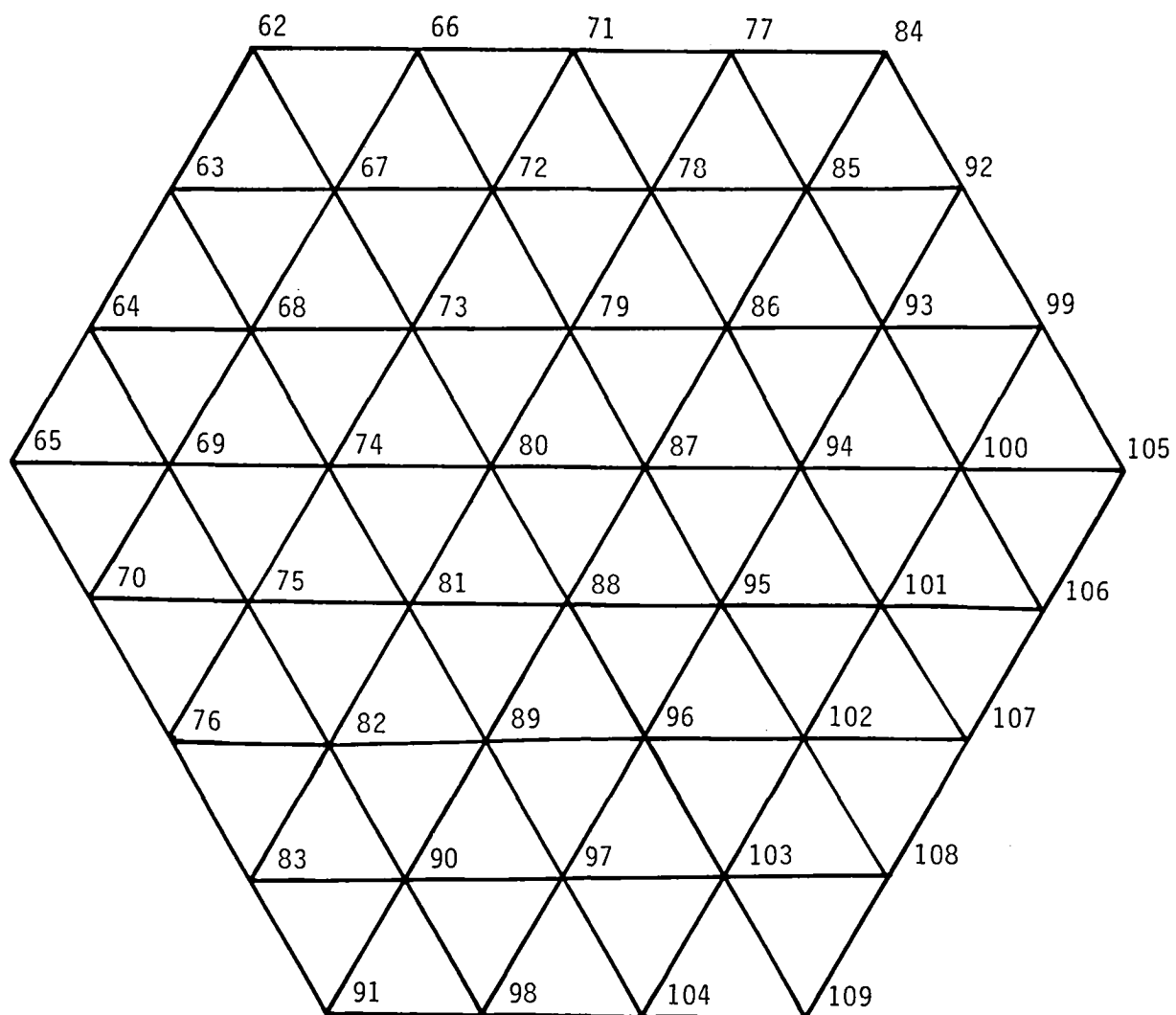
(a) Overall model.

Figure 8.- Finite-element model of radiometer-antenna reflector.



(b) Upper surface.

Figure 8.- Continued.



(c) Lower surface.

Figure 8.- Concluded.

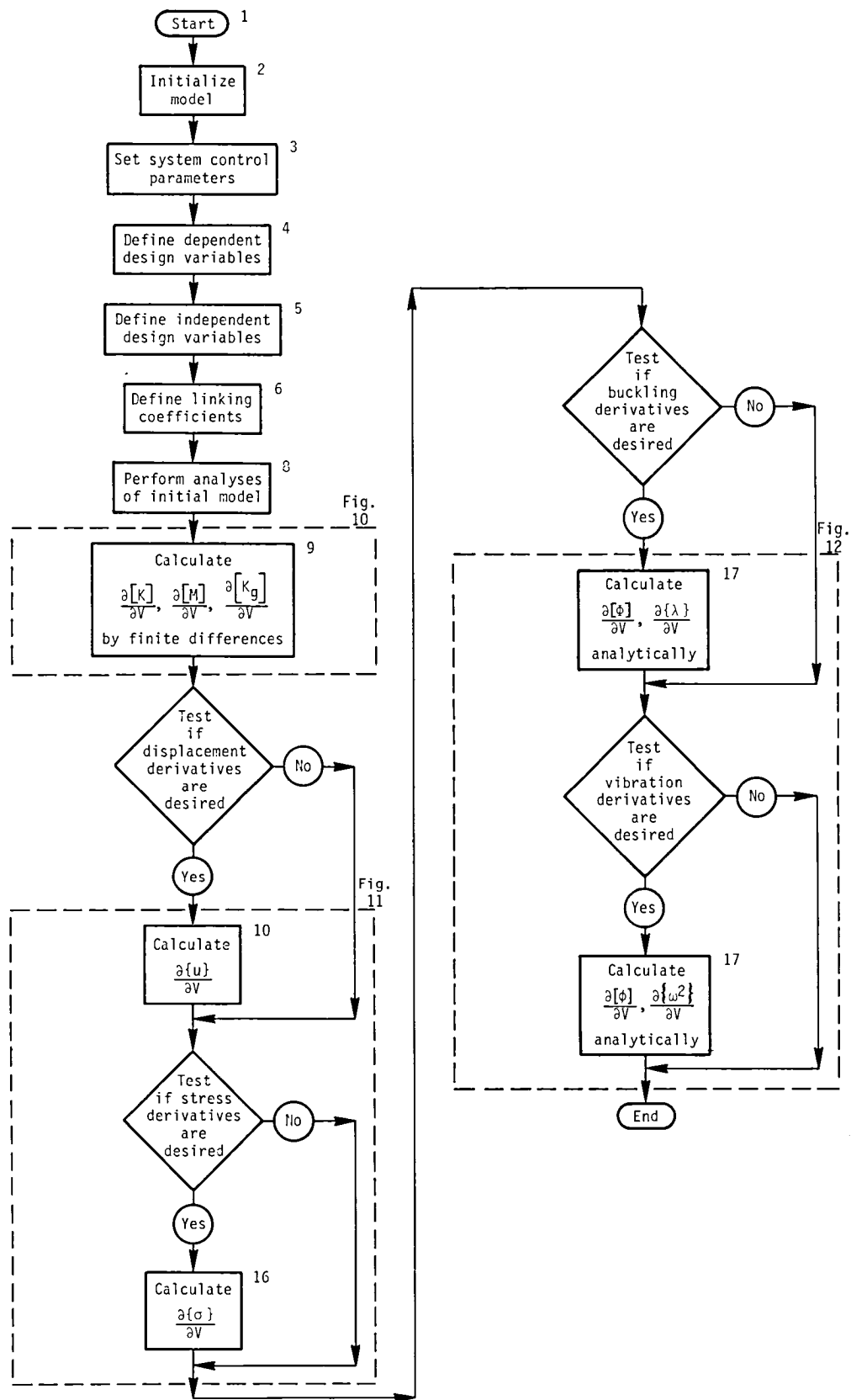


Figure 9.- Flowchart for system of runstreams to calculate structural-sensitivity derivatives with semianalytical method. Numbers indicate appropriate sections of runstreams in appendix.

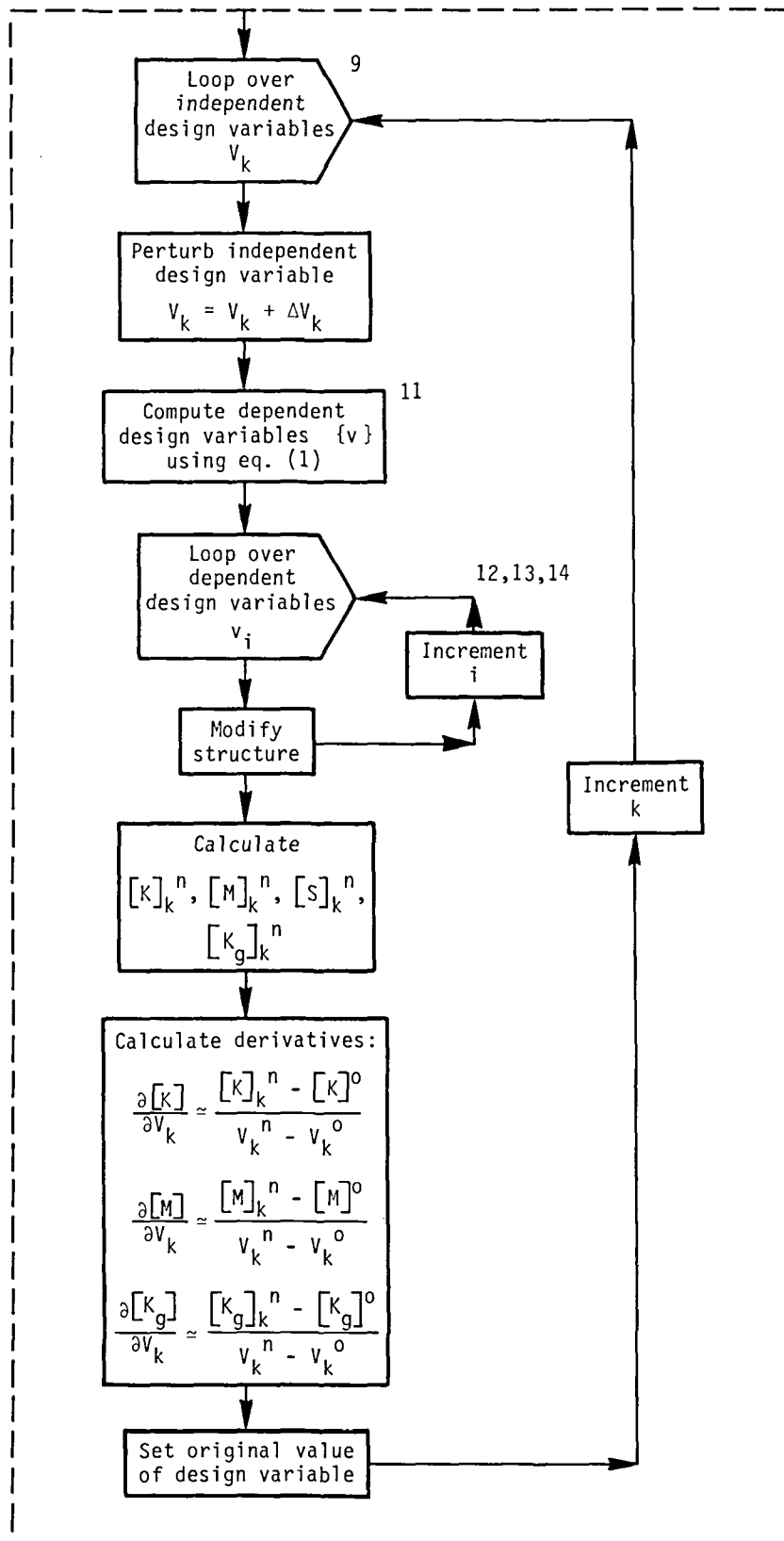


Figure 10.- Flowchart of section of runstream system which calculates derivatives of stiffness, mass, and geometric stiffness matrices. Numbers indicate appropriate sections of runstreams in appendix.

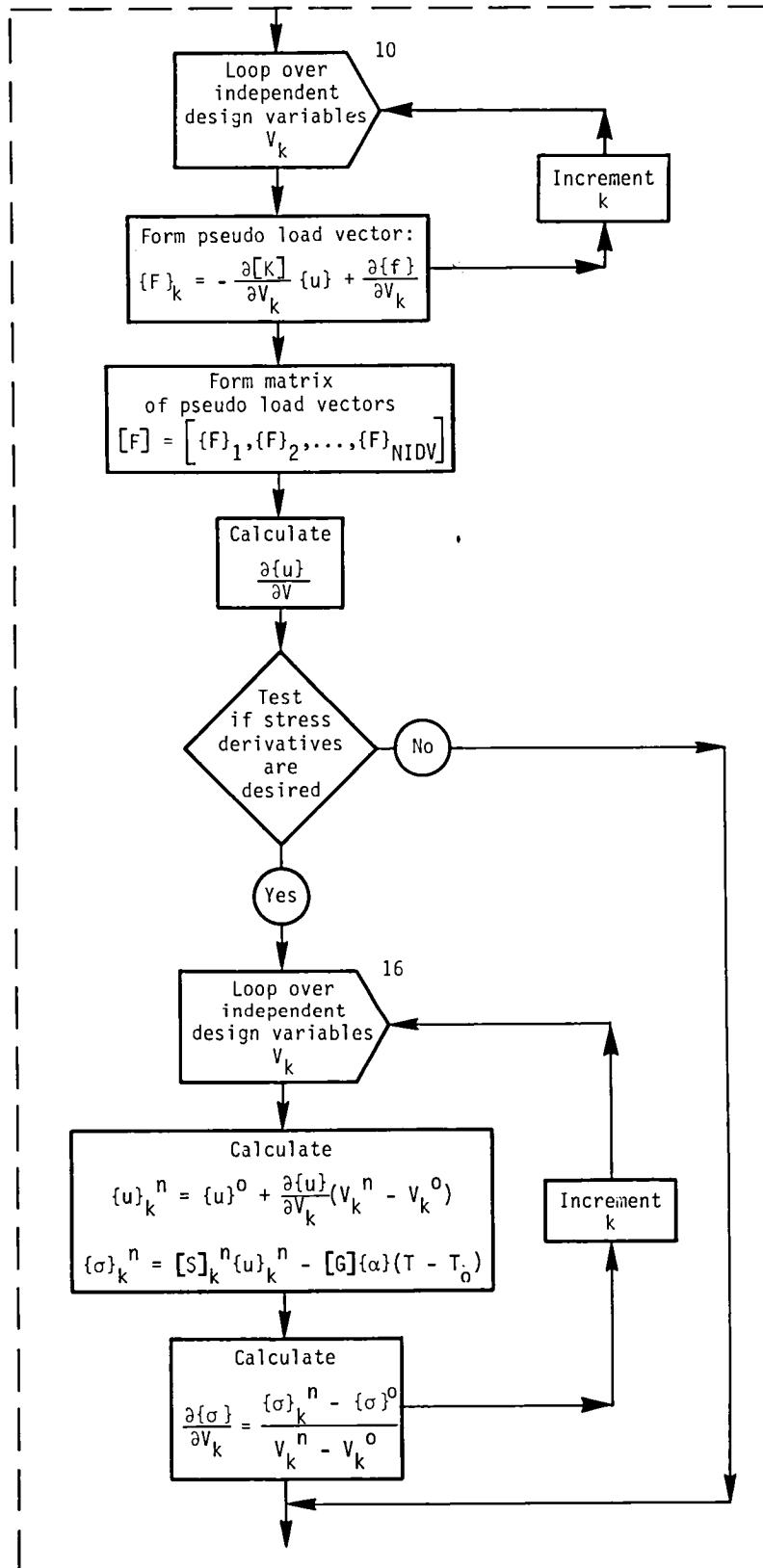


Figure 11.- Flowchart of section of runstream system which details calculation of displacement and stress derivatives. Numbers indicate appropriate sections of runstreams in appendix.

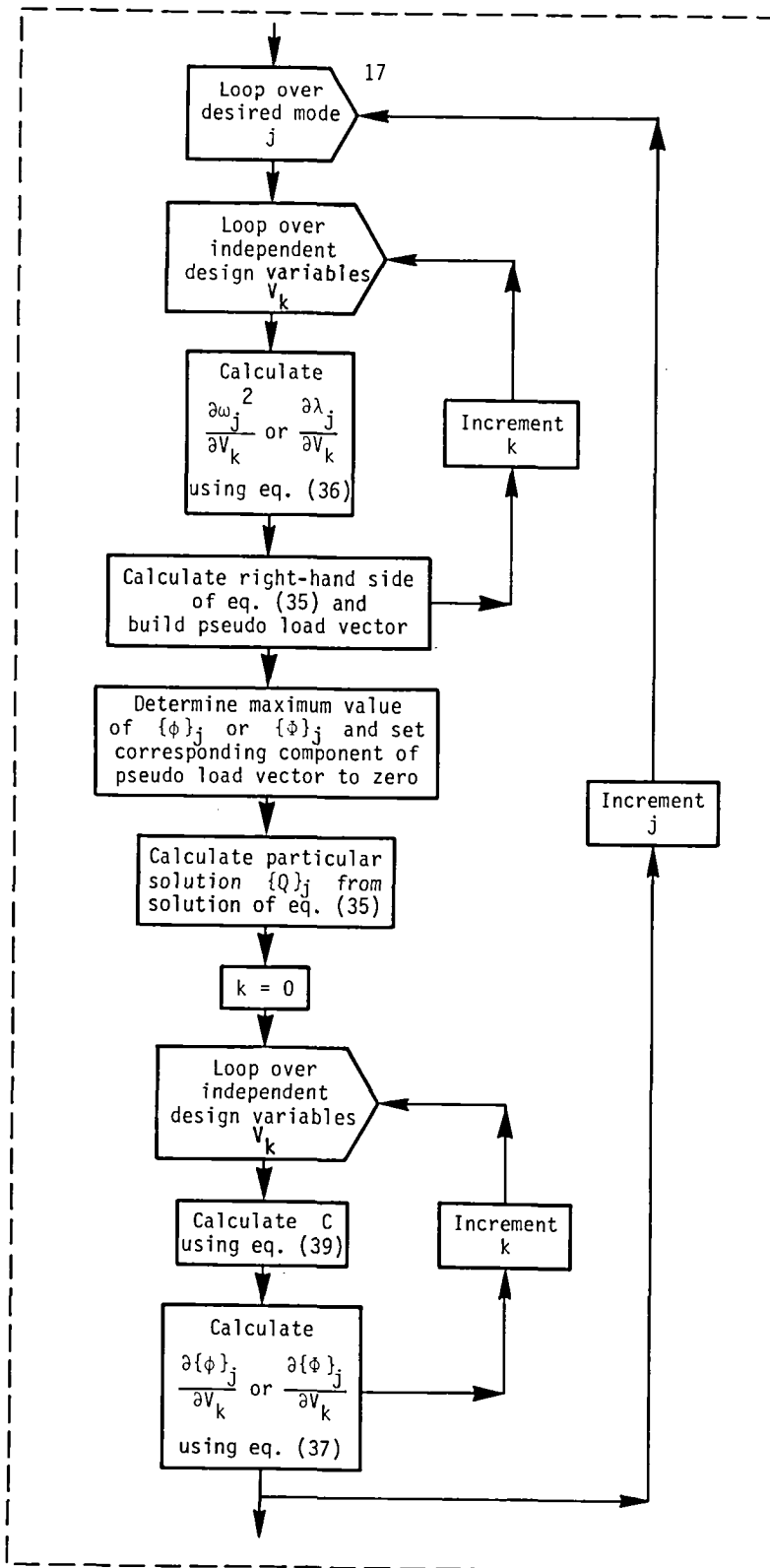


Figure 12.- Flowchart of section of runstream for calculation of derivatives of vibration and of buckling eigenvectors and eigenvalues. Numbers indicate appropriate sections of runstreams in appendix.

1. Report No. NASA TM-85743		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle STATIC AND DYNAMIC STRUCTURAL-SENSITIVITY DERIVATIVE CALCULATIONS IN THE FINITE-ELEMENT-BASED ENGINEERING ANALYSIS LANGUAGE (EAL) SYSTEM				5. Report Date March 1984	
				6. Performing Organization Code 506-53-53-07	
7. Author(s) Charles J. Camarda and Howard M. Adelman				8. Performing Organization Report No. L-15659	
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665				10. Work Unit No.	
				11. Contract or Grant No.	
				13. Type of Report and Period Covered Technical Memorandum	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546				14. Sponsoring Agency Code	
15. Supplementary Notes					
16. Abstract This paper describes the implementation of static and dynamic structural-sensitivity derivative calculations in a general purpose, finite-element computer program denoted the Engineering Analysis Language (EAL) System. Derivatives are calculated with respect to structural parameters, specifically, member sectional properties including thicknesses, cross-sectional areas, and moments of inertia. Derivatives are obtained for displacements, stresses, vibration frequencies and mode shapes, and buckling loads and mode shapes. Three methods for calculating derivatives are implemented (analytical, semianalytical, and finite differences), and comparisons of computer time and accuracy are made. Results are presented for four examples: a swept wing, a box beam, a stiffened cylinder with a cutout, and a space radiometer-antenna truss.					
17. Key Words (Suggested by Author(s)) Structural sensitivity analysis Large space antenna Structural optimization Finite-element method Thermal loads Engineering Analysis Language			18. Distribution Statement Unclassified - Unlimited Subject Category 39		
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 78	22. Price A05		

National Aeronautics and
Space Administration

Washington, D.C.
20546

Official Business

Penalty for Private Use, \$300

THIRD-CLASS BULK RATE

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA-451



NASA

POSTMASTER: If Undeliverable (Section 158
Postal Manual) Do Not Return
